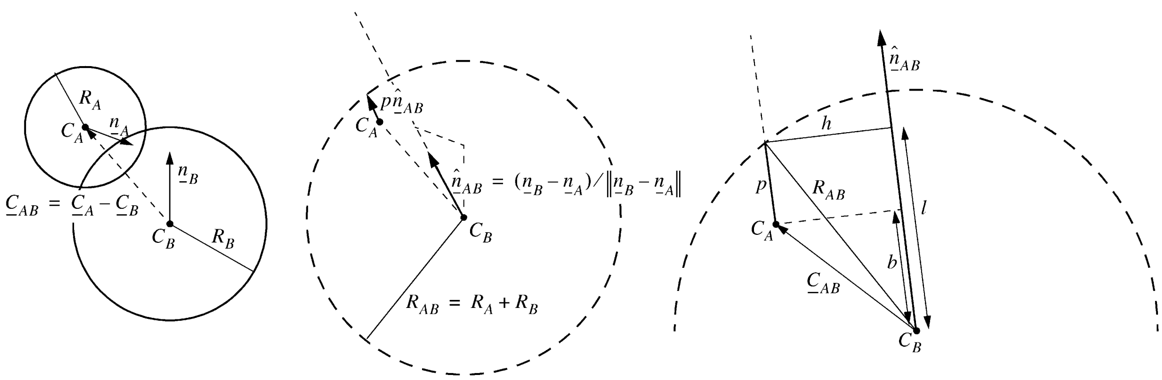
JRC TECHNICAL REPORTS

Implementation of Assembled Surface Normals and of a Penalty Contact Formulation in the Pinball Model of EUROPLEXUS

Folco Casadei
Martin Larcher
George Valsamos
Vincent Faucher

2014

Report EUR 26714 EN
JRC90939



European Commission
Joint Research Centre
Institute for the Protection and Security of the Citizen

Contact information
Martin Larcher
Address: Joint Research Centre, Via Enrico Fermi 2749, TP 480, 21027 Ispra (VA), Italy
E-mail: martin.larcher@jrc.ec.europa.eu
Tel.: +39 0332 78 9563
Fax: +39 0332 78 9049

http://ipsc.jrc.ec.europa.eu/
http://www.jrc.ec.europa.eu/

This publication is a Technical Report by the Joint Research Centre of the European Commission.

JRC90939
EUR 26714 EN
ISBN 978-92-79-38984-9
ISSN 1831-9424
doi:10.2788/88194

Luxembourg: Publications Office of the European Union, 2014

Printed in Italy

Contents

List of Figures

List of Tables

1. Introduction

The most popular contact algorithms available in finite element computer codes are probably the so-called slide line (in 2D) and slide surface (in 3D) algorithms proposed by Hallquist and Benson [1-2]. They are based on the notion of penetration of slave nodes into master segments (in 2D) or into master surfaces (in 3D). These algorithms suffer from a number of geometrically pathological cases in which physical penetration is not detected.

The pinball method proposed by Belytschko and co-workers from the late 80’s [3-10] for application in impact problems with penetration is much more robust as concerns penetration detection. The pinball contact-impact method has been implemented in EUROPLEXUS in [12-15], initially based upon a strong, Lagrange-multiplier based solution strategy of the contact constraints (see [11] for details of the method).

EUROPLEXUS [16] is a computer code for fast explicit transient dynamic analysis of fluid-structure systems jointly developed by the French Commissariat à l’Energie Atomique et aux Energies Alternatives (CEA Saclay) and by the Joint Research Centre of the European Commission (JRC Ispra).

Recently, the so-called Assembled Surface Normal (ASN) algorithm of Belytschko [3] and an alternative penalty-based solution of the contact constraints have also been introduced as an option in the code. They are described in the present report, which is organized as follows:

- Section 2 presents the contact-impact model based upon pinballs with Assembled Surface Normals (ASN) and using a penalty approach.
- Section 4 presents some implementation details.
- Section 5 presents some numerical examples for the validation of the newly implemented models.
- Appendix A contains the algorithm used to compute the closest points on two segments.
- Appendix B contains an unpublished (and slightly incomplete) paper which gives many details on the hierarchic pinball contact-impact model with Lagrange Multipliers.
- Finally, Appendix C contains a listing of all the input files mentioned in the present report.

2. Contact algorithm for continuum elements

The version of the pinball method based upon a penalty approach to compute contact forces is presented in this Section. We assume a mesh discretization consisting of (bulky) continuum elements, see Figure 1 for a 2D example, and an erosion criterion to eliminate failed elements from the calculation. Due to erosion, the active surface of the contacting bodies tends to become very irregular during penetration. Therefore, a method is needed to compute a relatively smoothly-varying normal to the surface. This normal should not follow the local variations due to irregular mesh boundaries, but represent an "average" normal to the surface. An algorithm called ASN (Assembled Surface Normal) to compute such a normal was proposed by Belytschko and Law in 1985 [3].

2.1 Assembled Surface Normals (ASN)

The algorithm consists of the following steps (see the sample mesh of Figure 1):

- Build element face normals, see Figure 2. These normals are oriented "outgoing" from the element and are *not* normalized, i.e. the norm of the vector equals the length (in 2D) or area (in 3D) of the face.
- Assemble element face normals at each element's nodes, to obtain element nodal normals, see Figure 3. These are not normalized.
- Assemble element nodal normals at global nodes, to obtain (global) nodal normals, see Figure 4. These are not normalized. Note that the resulting normals are zero (within round-off errors) at internal nodes, while they are non-zero at nodes on the surface of the mesh. Vanishing normals are set exactly to zero.
- Assemble nodal normals to obtain element normals, see Figure 5. These are not normalized.
- Normalize the resulting vectors to unit length, to obtain normalized element normals, see Figure 6.

As an alternative one could, after the first three steps of the previous procedure:

- Normalize the global nodal normals, see Figure 7.
- Assemble (the normalized) nodal normals to obtain (variant) element normals, see Figure 8. These are not normalized.
- Normalize the resulting vectors to unit length, to obtain normalized (variant) element normals, see Figure 9.

By comparing Figures 6 and 9 one sees that the final result depends upon the chosen procedure (basic or variant), but only slightly.

2.2 Calculation of the penetration

To apply the penalty method in a contact problem, it is necessary to compute the penetration between the two contacting bodies. In the pinball method, the penetration is computed for each couple of overlapping pinballs. A (parent, i.e. zero-level) pinball is embedded in each element having a non-zero normal as a result of the ASN procedure described at the previous paragraph. Normally such elements are all and only the elements having at least one face on the surface of the body (surface elements), at least if a sufficient spatial discretization of the problem is assumed.

A pathological case is illustrated in Figure 10, where some elements "on the surface" have a zero normal. This happens because there is only one layer of continuum elements across the body. This situation is not acceptable in the class of problems of interest here (impact problems with penetration discretized by continuum elements) and is therefore ignored in the following treatment.

Consider a generic couple of pinballs A, B which overlap partially, thus indicating the presence of contact, see Figure 11a. Let R_A, R_B be the radii and C_A, C_B the centres of the two pinballs. The vector joining the two centres is then:

$$\underline{C}_{AB} = \underline{C}_A - \underline{C}_B \tag{1}$$

and, since we have assumed that the two pinballs are overlapping, it is:

$$\|\underline{C}_{AB}\| < (R_A + R_B)\,. \tag{2}$$

We indicate by $\underline{n}_A$, $\underline{n}_B$ "the" normals to the two pinballs, i.e. the approximate normals to the two bodies in the elements to which the pinballs are attached, computed by the ASN procedure described above. Note that in both versions of the ASN procedure the final "element" normals are normalized and should therefore be of the same (unit) length. However, in Figure 11a, taken from reference [5], the two normals have different lengths (not normalized). This choice is more general, although in contrast with the ASN procedure as listed above and in the literature.

In order to study the interaction between the two pinballs it is useful to consider a fictitious "enlarged" pinball of center C_B and of radius equal to the sum of the radii:

$$R_{AB} = R_A + R_B\,. \tag{3}$$

This sphere, shown in Figure 11b, contains the B pinball entirely and its surface is the locus of the centres C_A of all pinballs A of radius R_A in contact with B with <u>zero overlap</u>.

2.2.1 Penetration direction

First, the penetration direction, i.e. the direction along which the penetration is assumed to occur, is defined as the "average" normal direction of the two pinballs (see Figure 11b), given by:

$$\hat{\underline{n}}_{AB} = (\underline{n}_B - \underline{n}_A) / \|\underline{n}_B - \underline{n}_A\| . \tag{4}$$

Note that this vector is normalized to unit length, as indicated by a superposed "hat".

Then, the penetration p is defined as the (relative) displacement of the two pinball centres along this direction, needed to exactly eliminate the overlap, as shown in Figure 11b by the vector $p\hat{\underline{n}}_{AB}$. The situation of Figure 11b is enlarged in Figure 11c for clarity.

2.2.2 Penetration

Then the penetration p can be computed as follows. The distance b is given by:

$$b = \underline{C}_{AB} \cdot \hat{\underline{n}}_{AB} , \tag{5}$$

the distance h is such that:

$$h^2 = \|\underline{C}_{AB}\|^2 - b^2 \tag{6}$$

and the distance l is given by:

$$l = \sqrt{R_{AB}^2 - h^2} . \tag{7}$$

Finally, the penetration p is given by:

$$\begin{aligned} p &= l - b = \\ &= \sqrt{R_{AB}^2 - h^2} - \underline{C}_{AB} \cdot \hat{\underline{n}}_{AB} = \\ &= \sqrt{R_{AB}^2 - \|\underline{C}_{AB}\|^2 + (\underline{C}_{AB} \cdot \hat{\underline{n}}_{AB})^2} - \underline{C}_{AB} \cdot \hat{\underline{n}}_{AB} \end{aligned} . \tag{8}$$

It is important to check that the expression given for the penetration is symmetric, i.e. that by exchanging the two pinballs A and B (and by denoting the resulting quantities by a superposed prime) the same value of p is obtained. This is readily verified. In fact, one has:

$$\begin{aligned} \hat{\underline{n}}_{BA} &= -\hat{\underline{n}}_{AB} \\ \underline{C}_{BA} &= -\underline{C}_{AB} \\ \|\underline{C}_{BA}\| &= \|\underline{C}_{AB}\| \\ R_{BA} &= R_{AB} \end{aligned} , \tag{9}$$

so that instead of (5-7) we obtain:

$$b' = \underline{C}_{BA} \cdot \hat{\underline{n}}_{BA} = b, \tag{10}$$

$$h'^2 = \left\| \underline{C}_{BA} \right\|^2 - b'^2 = h^2, \tag{11}$$

$$l' = \sqrt{R_{BA}^2 - h'^2} = l. \tag{12}$$

Finally one sees that:

$$p' = l' - b' = p. \tag{13}$$

2.3 Calculation of the rate of penetration

Some penalty methods involve not only the penetration p, but also the <u>rate of penetration</u> $\dot{p}$:

$$\dot{p} = \frac{dp}{dt}. \tag{14}$$

In reference [5], the following expression to compute $\dot{p}$ for a couple of contacting hexahedral elements (8 nodes each) A and B is given:

$$\dot{p}_{AB} = \frac{1}{8} \sum_{I=1}^{8} [(\underline{v}_{BI} - \underline{v}_{AI}) \cdot \hat{\underline{n}}_{AB}] \tag{15}$$

where I is the generic node of either A or B, $\underline{v}_I$ are the nodal velocities and $\hat{\underline{n}}_{AB}$ is the direction of penetration between the two pinballs, given by (4). This expression corresponds to assuming that “the” velocity associated with a pinball is the velocity at the centroid of the corresponding element. In fact, (15) can be re-written as:

$$\dot{p}_{AB} = \frac{1}{8} \left(\sum_{I=1}^{8} \underline{v}_{BI} \right) \cdot \hat{\underline{n}}_{AB} - \frac{1}{8} \left(\sum_{I=1}^{8} \underline{v}_{AI} \right) \cdot \hat{\underline{n}}_{AB}. \tag{16}$$

But the velocity at any point P in the hexahedron element can be interpolated as:

$$\underline{v}_P = \sum_{I=1}^{8} N_I \underline{v}_I \tag{17}$$

where N_I are the tri-linear shape functions:

$$N_I(\xi, \eta, \zeta) = \frac{1}{8}(1 - \xi_I \xi)(1 - \eta_I \eta)(1 - \zeta_I \zeta) \tag{18}$$

over the parent element and ξ, η, ζ are normalized coordinates. At the element centroid C it is $\xi = \eta = \zeta = 0$, so that from (18) $N_I(C) = 1/8$ and from (17) the velocity at the centroid is:

$$\underline{v}_C = \frac{1}{8} \sum_{I=1}^{8} \underline{v}_I. \tag{19}$$

Therefore, the expression (15) or (16) can be re-written as:

$$\dot{p}_{AB} = (\underline{v}_B - \underline{v}_A) \cdot \hat{\underline{n}}_{AB}, \tag{20}$$

where $\underline{v}_A$, $\underline{v}_B$ are the (interpolated) velocities at the centres of the two pinballs, which in this case coincide with the centroids of the corresponding elements. The right-hand side of (20) represents the relative velocity of the pinballs projected along the direction of penetration, and is indicated by the (signed) quantity v_{ABn} in the example of Figure 12. In conclusion, the rate of penetration is:

$$\dot{p}_{AB} = v_{ABn}. \tag{21}$$

Note that the penetration tends to increase when $v_{ABn} > 0$, while it tends to decrease (incipient rebound) when $v_{ABn} < 0$.

2.4 Interaction between a couple of pinballs

To summarize, the interaction between any couple of pinballs A, B is computed as follows.

- Compute the vector $\underline{C}_{AB}$ by (1):

$$\underline{C}_{AB} = \underline{C}_A - \underline{C}_B \tag{22}$$

- Compute the sum of the radii by (3):

$$R_{AB} = R_A + R_B \tag{23}$$

- Check whether the two pinballs are in contact by (2):

$$\|\underline{C}_{AB}\| < R_{AB} \tag{24}$$

- If the above inequality does not hold, then there is no interaction between these two pinballs: go to the next couple of pinballs. Else:

- Compute the penetration direction by (4):

$$\hat{\underline{n}}_{AB} = (\underline{n}_B - \underline{n}_A) / \|\underline{n}_B - \underline{n}_A\| \tag{25}$$

- Compute the distance b by (5):

$$b = \underline{C}_{AB} \cdot \hat{\underline{n}}_{AB} \tag{26}$$

- Compute the quantity h^2 by (6):

$$h^2 = \|\underline{C}_{AB}\|^2 - b^2 \tag{27}$$

- Compute the distance l by (7):

$$l = \sqrt{R_{AB}^2 - h^2} \tag{28}$$

- Finally, compute the penetration p by (8):

$$p = l - b \tag{29}$$

- If the rate of penetration is required by the chosen penalty model, then:
- Compute the velocities at the two pinball centers, $\underline{v}_A$, $\underline{v}_B$ by (17):

$$\underline{v}_A = \sum_{I=1}^{8} N_I(C_A)\underline{v}_{AI} \qquad ; \qquad \underline{v}_B = \sum_{I=1}^{8} N_I(C_B)\underline{v}_{BI} \tag{30}$$

- Compute the rate of penetration by (20):

$$\dot{p}_{AB} = (\underline{v}_B - \underline{v}_A) \cdot \hat{\underline{n}}_{AB} \tag{31}$$

It is interesting to consider how the penetration between two pinballs varies as a function of the relative positions of the two pinballs. This is illustrated in Figure 13. We assume two pinballs A and B of different radii, for generality. The relative normal $\hat{\underline{n}}_{AB}$ is assumed vertical in the example. The Figure shows the variation of the penetration $\underline{p} = p\hat{\underline{n}}_{AB}$ as the A pinball moves along a line normal to $\hat{\underline{n}}_{AB}$.

2.5 Calculation of the penalty force

When two pinballs A and B are in contact, a (repulsive) penalty force is applied. The expression given in reference [8] for the penalty force exerted by B on A is:

$$\underline{F}_{pA} = (k_p p + k_v \dot{p})\hat{\underline{n}}_{AB} \tag{32}$$

while, of course, an equal in modulus and opposite force is exerted by A on B:

$$\underline{F}_{pB} = -\underline{F}_{pA} = -(k_p p + k_v \dot{p})\hat{\underline{n}}_{AB}. \tag{33}$$

The direction of the force coincides with the direction of penetration $\hat{\underline{n}}_{AB}$, given by (4). The first term in the penalty force is proportional via a coefficient k_p to the penetration p given by (8), while the second term is proportional via a coefficient k_v to the rate of penetration $\dot{p}$ given by (20).

2.5.1 Classical expressions of penalty coefficients in 3D

In [8] the following expression is proposed for the coefficient k_p:

$$k_p = \frac{\beta K S^2}{V} \tag{34}$$

where K is the material's bulk modulus (see below), S is the "area of the impacted surface" and V is the volume of the element. The quantity β is a scaling ("tuning") coefficient (dimensionless number) which should be prescribed by the user. The same expressions are given also in reference [5].

Let us consider the 3D case, where the pinballs are spheres of radius R. By assuming that the area S is the cross-section of the pinball:

$$S = \pi R^2 \tag{35}$$

and that the volume V of the element is (at least approximately) equal to the volume of the pinball:

$$V = \frac{4}{3}\pi R^3 . \tag{36}$$

the expression (34) becomes:

$$k_p = \beta K R , \tag{37}$$

where the constant factor $(3\pi/4)$ has been incorporated in the β parameter, which has to be tuned anyway. If the interaction occurs between two pinballs of different materials (different bulk moduli) and/or with different radii, one should use in place of (37) the expression (taken from refs. [5, 8]):

$$k_p = \beta \frac{K_A R_A + K_B R_B}{2} \tag{38}$$

which accounts for the properties of both pinballs. Of course, (38) reduces to (37) if the two pinballs are identical in radius and material.

2.5.2 Penalty coefficients in 2D

In the 2D (plane) case, strictly speaking pinballs are circles. In this case in principle one would have in place of (35) and (36), respectively:

$$S = 2R \tag{39}$$

$$V = \pi R^2 \tag{40}$$

and (34) would become:

$$k_p = \beta K \tag{41}$$

where the factor $(4/\pi)$ has been incorporated in the β parameter, which has to be tuned anyway. This expression differs from (37) for the 3D case and is <u>dimensionally incorrect</u> because K has the dimensions of a pressure (N/m^2), see below, while the stiffness k_p should be expressed in N/m (see e.g. eq. 32).

Unfortunately, reference [8] only reports the 3D case and no mention is made of the 2D case. In addition, no expression is provided in [8] (nor in any other of the cited references) for the coefficient k_v. Therefore it is assumed that, if needed, k_v must be provided by the user.

The same expressions of the penalty force (32-38) are given also in reference [5] and in reference [6] (which, however, contains various evident typing errors in these formulas).

2.5.3 Classical expressions of penalty coefficients for shells

In reference [9], which deals with hierarchic pinballs mainly for contact between shell elements, different expressions from (32) are given for the penalty force:

$$\underline{F}_{pA} = F_{pA}\hat{\underline{C}}_{AB} = F_{pA}\frac{\underline{C}_{AB}}{\|\underline{C}_{AB}\|}, \qquad F_{pA} = \min(F_1, F_2) \tag{42}$$

where:

$$F_1 = \begin{cases} \dfrac{\rho_A\rho_B R_A^3 R_B^3}{\rho_A R_A^3 + \rho_B R_B^3}\dfrac{\dot{p}}{\Delta t} & \text{if } \dot{p} > 0 \\ 0 & \text{if } \dot{p} < 0 \end{cases} \tag{43}$$

$$F_2 = \left[\frac{K_A K_B}{K_A + K_B}\sqrt{\frac{R_A R_B}{R_A + R_B}}\right]p^{3/2} \tag{44}$$

where ρ_A, ρ_B are the densities of the two materials.

Note that according to reference [9], from the first of (42), the penalty force is always exerted in the direction between the pinball centers ($\hat{\underline{C}}_{AB}$), and not along the “penetration direction” $\hat{\underline{n}}_{AB}$ (4). This may be due to the fact that reference [9] uses hierarchic pinballs in shell elements. In fact, it is stated that “this choice enables to handle edge-to-surface, surface-to-surface and edge-to-edge contact”.

2.5.4 Expressions of penalty coefficients in EUROPLEXUS for sliding surfaces

Finally, in EUROPLEXUS the following expressions are implemented in subroutine CALPEN (called from FGLIS3, in turn called from GLIS3D) for contact computed with the so-called sliding surface algorithm [1-2] using the penalty method (input directive `LINK DECO GLIS PENA` ..., see User’s manual on page D2.180). This model is at the moment available only in 3D.

The contact penalty force includes only the term proportional to the penetration p:

$$\underline{F}_p = k_p p \hat{\underline{n}} \tag{45}$$

The contact stiffness k_p is computed as follows from the stiffness of master elements.

For a <u>continuum</u> master element:

$$k_p = \frac{\beta K S^2}{V} \tag{46}$$

(same expression as eq. 34), where β is a user-defined scaling factor, K is the material's bulk modulus (see below), S is the area of the contacting face (of the master element), and V is the volume of the master element. Note that in (34) S and V represent the cross-section and volume of the pinball, instead.

For a <u>shell</u> master element:

$$k_p = \frac{\beta K S}{L} \tag{47}$$

where β, K, S have the same meaning as in (46) and L is the maximum length of the master element's (shell) edges.

When contact occurs between two pinballs or sub-pinballs with different characteristics (bulk modulus, radius etc.), an expression such as (46) or (47) is evaluated for each of the pinballs obtaining thus two values k_{pA}, k_{pB}. Then the penalty coefficient is taken as the <u>minimum</u> of the two coefficients (*not* as an <u>average</u> like in expression 38):

$$k_p = \min(k_{pA}, k_{pB}). \tag{48}$$

This seems more physically intuitive than taking the average, since for a given inter-penetration of the two pinballs the contact force is clearly determined by the "weakest" of the two bodies, not by an average of the two material properties.

2.6 Bulk modulus

The bulk modulus K, or modulus of volume expansion, of a material subjected to a uniform (hydrostatic) pressure (P) field ($\sigma_x = \sigma_y = \sigma_z = -P$ and $\tau_{xy} = \tau_{yz} = \tau_{zx} = 0$) is the coefficient that relates the pressure P (with the minus sign to conform to the usual stress rule for compression) to the <u>volume expansion</u> $e = \varepsilon_x + \varepsilon_y + \varepsilon_z$:

$$-P = Ke \tag{49}$$

Therefore one can say that the bulk modulus characterizes the resistance of a material to a uniform pressure.

For an elastic material of Young's modulus E and Poisson's coefficient ν the normal strains are related to the normal stresses by:

$$\begin{aligned} \varepsilon_x &= \frac{1}{E}[\sigma_x - \nu(\sigma_y + \sigma_z)] \\ \varepsilon_y &= \frac{1}{E}[\sigma_y - \nu(\sigma_x + \sigma_z)] \\ \varepsilon_z &= \frac{1}{E}[\sigma_z - \nu(\sigma_x + \sigma_y)] \end{aligned} \tag{50}$$

By adding up these three equations and by denoting the sum of normal stresses $\Theta = \sigma_x + \sigma_y + \sigma_z$ one has:

$$e = \frac{1 - 2\nu}{E}\Theta \tag{51}$$

Under a hydrostatic pressure P field it is $\Theta = -3P$ and (51) becomes:

$$e = -\frac{3(1 - 2\nu)}{E}P \tag{52}$$

so that by comparing (49) and (52) one obtains the following expression for the <u>bulk modulus</u>:

$$K = \frac{E}{3(1 - 2\nu)} \tag{53}$$

The bulk modulus has the same units as Young's modulus, stress and pressure: Pa (i.e. N/m^2) in the standard system.

2.7 Distribution of penalty force on the element nodes

The penalty force (32) or (42) has to be distributed among the element's nodes. In reference [5], which deals with 0-level pinballs embedded in 8-node hexahedra, the following formula is proposed:

$$\underline{F}_{pI} = \frac{1}{8}\underline{F}_p \qquad I = 1, \ldots, 8, \tag{54}$$

i.e., the force is equally distributed among all the nodes I of the element. An alternative possibility is mentioned, whereby the force would be distributed only among the nodes of the impacted face of the element. The same formula is proposed also in reference [8].

In reference [6] a similar formula is proposed, but the force is (equally) distributed among "the four nodes of each element which are closest to the center of the other element". This is because hexahedral element faces have four nodes.

In reference [9], which deals with hierarchic pinballs mainly for shell elements, the penalty force (given by eq. 42) is distributed onto all the element's nodes (in this case a face has the same nodes as the whole element) according to the element's shape functions.

2.8 Considerations on penalty force between contacting elements or pinballs

Consider a couple of contacting (regular) hexahedral elements in 3D, see Figure 14. Let S denote the contact surface (in this case the area of the element face), h the element side and $V = Sh$ the element volume.

The contact stiffness according to (46) is:

$$k_p = \frac{\beta K S^2}{V} = \beta K S \frac{S}{V} = \frac{\beta K S}{h} \tag{55}$$

The contact force for a penetration p is therefore:

$$F_p = k_p p = \beta K S \frac{p}{h} \tag{56}$$

But p/h is the (engineering) strain in one of the elements in the direction of penetration:

$$\frac{p}{h} = \varepsilon_h \tag{57}$$

so that (56) can be re-written:

$$F_p = \beta K \varepsilon_h S \tag{58}$$

From (49) we see that the term $K\varepsilon_h$ is proportional (through a coefficient which can be included in the parameter β) to the (engineering) stress σ_h in the element along the direction of penetration (assuming a linear elastic material behavior):

$$K\varepsilon_h \approx \sigma_h \tag{59}$$

so that (58) becomes:

$$F_p = \beta \sigma_h S \tag{60}$$

From this expression we see that the contact penalty force is proportional to the force $(\sigma_h S)$ that would arise in the element when subjected to a displacement equal to the penetration p on one of its faces, assuming a linear elastic material.

From these considerations and from physical intuition it appears therefore that the value of the parameter β should be of the order of 1. The value assumed by default in the code is 1.

Consider now the same couple of contacting elements as before, but let the contact be described by (zero-level) pinballs, see Figure 15. The contact stiffness is given by (37):

$$k_p = \beta K R \tag{61}$$

where R is the radius of the pinballs ($R \approx h/2$). This, multiplied by the penetration p, gives the contact force which must then be distributed on the nodes. By default the contact force is assumed to act at the pinball centre, so it is distributed equally on the 8 nodes of the element. Another possibility, shown in Figure 15, is to apply the force at the center of the contacting face and therefore distribute it only among the four nodes of the face.

Finally, let us examine the case of hierarchic pinballs at a generic level L, see Figure 16. Recall that the parent pinball has level 0, and that at each successive refinement the radius of the pinballs is divided by 2. In the example a level $L = 2$ has been assumed.

In the case of "flat" "perfect" contact shown in the Figure, there are 2^L contacting pinball couples along each spatial direction of the (quadrilateral) face of the hexahedron, i.e. 2^{2L} contacts altogether on the 3D quadrilateral face, i.e. 16 contacts in the example (of which only the four "frontal" ones are visible).

Intuitively, for a given penetration the total contact force on the face should be (roughly) the same, irrespective of the chosen hierarchy level of the pinballs (in fact, hierarchic pinballs are used only to make contact detection more precise geometrically). This force is indicated as F_{p0} in Figure 16 at the zero-level (just one contact). Since for a quadrilateral face at level L there are 2^{2L} contacts, the single contact force has to be:

$$F_{pL} = \frac{1}{2^{2L}} F_{p0} \tag{62}$$

Since the penetration does not depend upon the level, it is the contact stiffness (61) that has to be scaled by the number of contacts:

$$k_{pL} = \frac{1}{2^{2L}} k_{p0} \tag{63}$$

Since the pinball radius varies with the level according to:

$$R_L = \frac{R_0}{2^L} \tag{64}$$

one finds that for each single pinball at level L of an impacting quadrilateral face the penalty contact force is given by:

$$F_{pL}^{\text{quad}} = k_{pL}^{\text{quad}} p = \frac{1}{2^L} \beta K R_L p \,. \tag{65}$$

At zero-level $2^L = 1$ and one recovers the expression for the parent pinball, see eq. (61).

Similarly, for an impacting triangular face at level L the number of descendent pinballs (i.e. the number of contacts) is $2^L(2^L + 1)/2$. Therefore:

$$F_{pL}^{\text{tria}} = k_{pL}^{\text{tria}} p = \frac{2}{2^L + 1} \beta K R_L p \,. \tag{66}$$

A similar expression remains to be derived for the 2D case.

2.9 Treatment of rebound with the penalty method

The treatment of rebound in the description of an impact problem by the penalty method deserves some attention.

When using the Lagrange Multipliers method, rebound must be specifically addressed by an *ad hoc* strategy. In reference [13] two alternatives are considered: the so-called *a-priori* rebound or the *a-posteriori* rebound.

- In the first case, which is also by default the one chosen in the code, the relative penetration velocity of the two pinballs is estimated and, depending on the sign that this quantity assumes, the contact constraint is or is not enforced. The constraint is enforced when the penetration tends to increase, but is not enforced when the penetration starts to diminish (incipient rebound).
- In the second case, a constraint is always retained as long as penetration occurs, but the sign of the contact forces is inspected after solution of the system of constraints and, if the force is attractive, then the force is set to zero. This method works in very simple cases where each contact constraint is independent from any other constraints, but it may fail in more complex situations, and therefore the *a priori* method is preferred (and is assumed by default).

The type of rebound algorithm can be chosen by an input option: `OPTI PINS REB1` chooses the a priori rebound (which is usually redundant since this is the default). `OPTI PINS REB2` chooses the a posteriori rebound. Finally, `OPTI PINS NORB` completely disables the rebound treatment.

When using the penalty approach, it is necessary to **let the contact (penalty) force act as long as there is penetration, irrespective of the relative velocity of the pinballs**. In other words, we do not want any special treatment of rebound with penalty. Therefore, any rebound-related options (`REB1`, `REB2`, `NORB`) are simply ignored if the penalty approach for pinball contact is specified.

2.10 Generalized ASN algorithm for hierarchic pinballs

In a previous Section (2.1) the ASN algorithm proposed by Belytschko and Law in 1985 [3] has been recalled for the case of zero-level (parent) pinballs. In their paper on hierarchic pinballs [9] (mostly focusing on contacts between shells) Belytschko and Yeh use the ASN algorithm only to find the external faces of the domain. In fact, in their implementation penalty forces are always directed along the line that joins the centers of the two contacting pinballs.

Here we want to explore the possibility of using a generalized ASN concept also for hierarchic pinballs, both with continuum and with shell elements. The original ASN algorithm as described above will be used only for contact between 0-level pinballs, i.e. in applications without hierarchic pinballs (e.g. perforation of bulky structures with erosion), which typically do not require a high precision of contact detection anyway, and in which too precise a representation of the eroded surface would probably be a drawback rather than an advantage.

2.10.1 Single element

Consider first a single (isolated) 2D quadrilateral, as shown in Figure 17. At zero level (left part of the Figure), the standard ASN algorithm yields the elemental face normals shown in red, the nodal normals shown in green, and the element (or pinball) assembled normal shown in black (at the pinball center). The element normal vanishes in this case and is therefore not visible.

Passing to refinement level 1 (central part of the Figure), four descendent pinballs are created. Each of these is adjacent to a "vertex" of the element, and therefore they are called vertex pinballs (marked by a V on the drawing). It seems natural that a vertex pinball has associated the nodal normal of the corresponding vertex, as shown in the Figure.

Passing to refinement level 2 (right part of the Figure), twelve descendent pinballs are created. Of these, four are vertex pinballs (V) while the others are face pinballs (F) because they are adjacent to a face of the parent element. It seems natural that a face pinball has associated the element face normal of the corresponding face, as shown in the Figure. Note that, upon refinement, only descendent pinballs which are in contact with an external face of the element are generated in the recursive pinball splitting process: "internal" descendent pinballs are not generated. This refinement procedure could go on at will, with the same strategy.

2.10.2 Simple mesh

Now consider a simple mesh of 4-node quadrilaterals as shown in Figure 18. At zero-level, there are four (parent) pinballs, one for each element of the mesh. These pinballs now look like vertex pinballs (with respect to the mesh) and have non-zero ASN normals, which happen to coincide with the normals at the corresponding vertices of the mesh. At refinement levels 1 and 2 the resulting descendent pinballs and the associated normals are as shown in the central and right parts of Figure 18, respectively.

2.10.3 Flat aligned contact

The case of "flat" "aligned" contact is shown in Figure 19. For "internal" contacting pairs of pinballs the penetration direction is normal to the interface, as expected. It is important to note that the same holds also for contacting pairs at the vertices, despite the fact that the pinball normals have a certain inclination at vertices.

2.10.4 Flat mis-aligned contact

In the case of moderate mis-alignment of the pinballs (by less than a pinball radius) as shown in Figure 20, the algorithm continues to give the same penetration directions as in the perfectly aligned case.

2.10.5 Vertex-to-face contact

The case of vertex-to-face contact is illustrated in Figure 21. The standard ASN formula for the calculation of the penetration direction gives a vector which is reasonably normal to the face. Alternatively, one can define a rule stating that, in case of contact between a vertex pinballs and a face pinball, the penetration direction is the normal to the face rather than an average of the two normals.

2.11 Contact rules

On the basis of the considerations detailed in the previous Sections, we tentatively define a set of rules for the contact between bodies by the (basic or hierarchic) pinballs method. These rules exploit the concept of an Assembled Surface Normal (ASN) associated (whenever this makes sense) with each parent and with each descendent pinball. They are developed primarily for use with a contact enforcement formulation based on penalty forces. However, their use is not excluded also in a Lagrange multipliers context. The advantage of penalty with respect to Lagrange multiplier (LM) formulations is that they avoid all the problems related with contact redundancies, which lead to ill-conditioned matrices in the LM method.

2.11.1 ASNs and pinball types

To each parent pinball is associated a unique Assembled Surface normal (ASN), obtained with (a variant of) the original algorithm by Belytschko and Law of Section 2.1 and representing an (average) normal direction to the body surface in the associated finite element e. The exact implementation of the ASN algorithm is described below in Section 2.11.2. Tentatively, such normal is normalized, i.e. it is represented by a vector of unit length $\hat{\underline{n}}_e$—except in the case where a normal cannot be defined, in which all normal components are set to 0. This happens when the assembly process yields a vanishing resultant vector, i.e. for the pathological configuration of Figure 10, for an “isolated” finite element (not connected to any other elements) and, finally, for a material point (single-node element).

Also to each descendent pinball is associated a unique normal (always normalized to unit length, since zero-length normals cannot occur for descendent pinballs), whose calculation is detailed below in this paragraph, according to the “type” of pinball introduced next.

Therefore, the ASN associated with a parent pinball (E pinball, see below) can have either length 1.0 or length 0.0 (when it is undefined). Instead, the ASN associated with a descendent pinball (F, C or V, see below) is always defined and therefore it always has length 1.0.

Types of pinballs

In addition, each pinball (parent or descendent) has an associated type: “element” (E), “face” (F), “corner” (C) or “vertex” (V).

Element (E) pinballs are parent ($L = 0$) pinballs. They are associated with an (entire) element.

The other pinball types (F, C, V) are descendent ($L > 0$) pinballs and are associated with a portion of an element.

As shown in Figure 28 for the various cases both in 2D and in 3D, F pinballs are located along an (external) face. C pinballs occur only in 3D and are located along an (external) corner, the intersection line between two (external) element faces. Finally, V pinballs are adjacent to a vertex of the element, the intersection point (mesh node) of two (external) element faces in 2D, or of three (external) element faces in 3D.

Both the ASN and the type of each pinball have to be re-computed at each time step because of changes in the mesh configuration (large motions and large strains) and in the connectivity (erosion, adaptivity). In particular, when an element fails and is eroded some previously internal element faces become external, some new pinballs have to be created on these new external surfaces and all the neighboring nodal normals, ASNs and pinball types are affected.

ASNs for the various pinball types

For an F pinball, the associated normal is the (unit) normal $\hat{\underline{n}}_F$ to the corresponding face. The calculation of these normals is straightforward for 2D faces and for triangular 3D faces, which are always planar. In the case of a quadrilateral 3D face, which may present some warping, "the" normal is defined here as the vector product of the two medians of the face, which intersect each other at the element centroid.

For a V pinball, the associated normal is the (unit) normal $\hat{\underline{n}}_V$ to the corresponding vertex, which coincides with a node of the mesh. These normals are computed and used to determine the ASN and are therefore readily available.

For a C pinball, the associated normal is the (unit) normal to the corresponding corner $\hat{\underline{n}}_C$. This could in principle be obtained by assembling and normalizing the normals to the two 3D (external) faces which form the corner under consideration. However, this task is complicated, in general, because the two external faces can belong to different elements (and therefore to different parent pinballs). To simplify this task, the algorithm tentatively implemented exploits the knowledge of the vertex normals. The two vertices of the corner are identified and the average of the two vertex normals is built:

$$\underline{n}_a = \hat{\underline{n}}_1 + \hat{\underline{n}}_2 . \tag{67}$$

Then, this vector is projected onto the plane normal to the corner (to eliminate any out-of-plane components), and finally it is normalized to unit length.

The (unit) tangent vector, directed along the corner of vertices $\underline{x}_1$, $\underline{x}_2$ is given by:

$$\hat{\underline{t}} = \frac{\underline{x}_2 - \underline{x}_1}{\|\underline{x}_2 - \underline{x}_1\|} . \tag{68}$$

The component of the average normal along the tangent is the following vector:

$$\underline{n}_{at} = (\underline{n}_a \cdot \hat{\underline{t}})\hat{\underline{t}} . \tag{69}$$

Finally, the unit normal to the corner is:

$$\hat{\underline{n}}_c = \frac{\underline{n}_a - \underline{n}_{at}}{\|\underline{n}_a - \underline{n}_{at}\|} . \tag{70}$$

For an E pinball, the associated normal is the corresponding ASN. This normal can have either unit or zero length. Instead, F, V and C pinball normals always have unit length.

2.11.2 Implementation of the ASN algorithm

The implemented version of the ASN algorithm is as follows.

- Build external element face normals, i.e. build face normals *only along external faces of the mesh*, see Figure 22 and comments at the end of this paragraph. These normals are oriented “outgoing” from the element and are *not* normalized, i.e. the norm of the vector equals the length (in 2D) or area (in 3D) of the face.
- Assemble external element face normals at each element’s nodes, to obtain (directly) global external nodal normals, see Figure 23. These are not (yet) normalized.
- Assemble nodal normals to obtain element normals, see Figure 24. These are not normalized.
- Normalize the resulting element normals to unit length, to obtain normalized element normals, see Figure 25.
- Normalize the nodal normals to unit length to obtain normalized nodal normals, see Figure 26.

It is important to note that the normalization of the nodal normals is performed last, i.e. *after* the use of these normals to obtain the element normals (which are also normalized after assembly). The example of Figure 27, consisting of a stand-alone triangular element, illustrates the two alternatives. With the algorithm as described above (top part of the Figure), the resulting element normal is zero (as expected, because the element is not connected to any other elements). Instead, if one would normalize the nodal normals before assembling them one would obtain a non-zero element normal (which seems unreasonable for a stand-alone element), as shown in the bottom part of the Figure.

Note that the fact that face normals are built only along the external faces, and not also along the internal ones like in the original implementation of the ASN algorithm by Belytschko and law (Section 2.1) is due to two reasons. First, external and internal faces are already identified for other purposes in the code and are readily available, so it would be a waste of CPU to compute and assemble normals at internal faces/nodes. Second, in the present implementation the ASN and in particular the occurrence of a vanishing normal is not used as a means of detecting internal nodes or faces. Vanishing ASNs or, more precisely, vanishing pinball normals can still occur, but they indicate either a material point or an isolated (stand-alone) continuum element.

2.11.3 Penetration direction

Two pinballs A, B penetrate each other if, according to (2):

$$\|\underline{C}_{AB}\| < R_A + R_B . \quad (71)$$

When this occurs, the penetration direction $\hat{\underline{n}}_{AB}$ can in principle be computed as follows (but for the algorithm actually implemented in the code see Section 4.1):

- If $\|\underline{n}_A\| = \|\underline{n}_B\| = 0$, i.e. if both pinballs are (element) pinballs with an undefined associated normal, then the penetration direction is the line joining the two pinball centres (see Figure 29):

$$\hat{\underline{n}}_{AB} = \underline{C}_{AB} / \|\underline{C}_{AB}\| \tag{72}$$

- Else, if ($\|\hat{\underline{n}}_A\| = 1$ and $\|\underline{n}_B\| = 0$), or if ($\|\underline{n}_A\| = 0$ and $\|\hat{\underline{n}}_B\| = 1$), i.e. if only one of the two pinballs has a defined normal, then the penetration direction coincides with either $-\hat{\underline{n}}_A$ or $\hat{\underline{n}}_B$, respectively, see Figure 30. This can be treated either as a special case, or by applying the general equation (4), since the result is the same.
- Else $\|\hat{\underline{n}}_A\| = 1$ and $\|\hat{\underline{n}}_B\| = 1$, i.e. both pinballs have a defined (unit) normal. Then the types of the two pinballs must be considered:
  - If both pinballs are E pinballs, then the penetration direction is computed with the general expression (4):

$$\hat{\underline{n}}_{AB} = (\underline{n}_B - \underline{n}_A) / \|\underline{n}_B - \underline{n}_A\| \tag{73}$$

  - Else if one pinball is an E pinball and the other is not an E pinball, then the normal associated with the other pinball (F, C or V) “has the precedence” over this one and determines the penetration direction. This case occurs when the E pinball is a parent (level $L = 0$) pinball associated either with a continuum or with a structural (shell/beam etc.) element (but not to a material point, since here the normal is assumed to be non-zero), while the other pinball (F, C or V) is necessarily a descendent pinball, i.e. at a level $L > 0$.
  - Else neither A nor B is an E pinball. Then:
    * If both pinballs are face pinballs, then the penetration direction is computed with the general expression (4), see Figure 31.
    * Else if one pinball is a face pinball and the other is a corner or vertex pinball, then the normal associated with the face pinball “has the precedence” over the other one and determines the penetration direction, see Figure 32.
    * Else if both pinballs are corner pinballs, then the penetration direction is computed with the general expression (4).

* Else if one pinball is a corner pinball and the other is a vertex pinball, then the normal associated with the corner pinball “has the precedence” over the other one and determines the penetration direction.

* Else both pinballs are vertex pinballs. Then, the penetration direction is computed with the general expression (4).

The situation in the case that both pinballs have a defined (unit) normal is summarized in the following Table, where G represents the general expression (4) or (73). In abscissa is represented the type of the first pinball and in ordinate the type of the second pinball (obviously, the table is symmetric with respect to the diagonal).

	E	V	C	F
E	G	V	C	F
V	V	G	C	F
C	C	C	G	F
F	F	F	F	G

Table 1 - Summary of normal calculations in the contact between two pinballs with unit normal.

2.12 Implementation of the contact force calculation

The implementation of the penalty contact force evaluation is detailed hereafter. As anticipated in Section 2.8, physically the contact force is given by an expression of the general form (60):

$$F_p = \beta\sigma_h S \tag{74}$$

where:

- β is the penalty coefficient (which should be roughly of the order of 1),
- σ_h is the normal stress generated by the penetration,
- S is the contact surface.

By assuming for simplicity a linear elastic material, the normal stress generated by the penetration can be approximated as (see also eqs. 57 and 59):

$$\sigma_h \approx K\varepsilon_h = K\frac{p}{h} \tag{75}$$

where:

- K is the material's bulk modulus (close to Young's modulus E, see eq. 53),
- ε_h is the engineering strain in the penetration direction,
- p is the penetration,
- h is the element's height in the penetration direction.

By combining (74) and (75) one gets (56):

$$F_p = \beta K \frac{S}{h} p, \qquad (76)$$

from which one sees that the force F_p will be expressed in N if β is a non-dimensional coefficient, K is expressed in Pa, S in m^2 and h and p in m.

2.12.1 3D continuum

In the case of contact between 3D continuum elements:

- S (contact surface) can be taken approximately equal to the pinball's (parent or descendent) cross-section (with R the pinball's radius):

$$S \approx \pi R^2 \qquad (77)$$

- h (height of the element) can be take approximately equal to the parent pinball's diameter, i.e. twice the parent pinball's radius R_0:

$$h \approx 2R_0 \qquad (78)$$

Then (76) becomes:

$$F_p = \beta K \frac{\pi R^2}{2R_0} p. \qquad (79)$$

Note that this formula is valid both for parent and for descendent pinballs and does not require any adjustment for the pinball level. In fact, the level is included in the S quantity, which depends upon the descendent's radius R.

If L is the level of the descendent pinball ($L \geq 0$), then it is approximately (see 64):

$$R \approx \frac{R_0}{2^L} \qquad (80)$$

so that:

$$R^2 \approx \frac{R_0^2}{2^{2L}} \qquad \text{or} \qquad \frac{R^2}{R_0} \approx \frac{R_0}{2^{2L}} \tag{81}$$

and (79) can also be written as:

$$F_p = \beta K \frac{\pi}{2} \frac{R_0}{2^{2L}} p \,. \tag{82}$$

Alternatively, one can express R_0 as a function of R from (80):

$$R_0 \approx 2^L R \qquad \text{or} \qquad \frac{R^2}{R_0} \approx \frac{R^2}{2^L R} = \frac{R}{2^L}, \tag{83}$$

so that (79) becomes also:

$$F_p = \beta K \frac{\pi}{2} \frac{R}{2^L} p \qquad \text{3D continuum} \tag{84}$$

The expression (84) is more efficient than (82) because it does not require the evaluation of the parent pinball's radius R_0.

Note that these formulas (82) or (84) are valid for 3D continuum elements <u>whatever the shape of their faces</u> (i.e. both for quadrilateral and for triangular faces).

2.12.2 2D continuum

In the case of contact between 2D continuum elements, one should distinguish between the plane stress / plane strain case and the axisymmetric case.

In the <u>plane strain or plane stress</u> case, let t be the thickness of the element in the direction normal to the plane (usually it is taken $t = 1$). The contact force is still given by (74) and the stress is still given by (75), so that expression (76) is still valid.

However, now for the contact surface S we have, in place of (77):

$$S \approx 2Rt \tag{85}$$

with R the pinball radius (parent or descendent). The height h of the element is still approximately equal to the parent pinball's diameter (see eq. 78), so that from (76) we obtain, in place of (79):

$$F_p = \beta K \frac{Rt}{R_0} p \,. \tag{86}$$

This expression is dimensionally correct because, if R, t, R_0 and p are expressed in m and K is expressed in Pa, then F_p results expressed in N.

The previous relations (80), (81) and (83) between R and R_0 are valid also in 2D, so that eq. (86) can be re-written as:

$$F_p = \beta K \frac{t}{2^L} p \qquad \text{2D plane stress / plane strain continuum} \tag{87}$$

In the 2D axisymmetric case, the same expressions derived above for the plane strain / plane stress case are valid, if one replaces the (constant) element thickness t by the (local) azimuthal thickness ("mean" radius) r_M, which is in this case given by the x-coordinate of the pinball's center x_C:

$$F_p = \beta K \frac{x_C}{2^L} p \qquad \text{2D axisymmetric continuum} \tag{88}$$

2.13 New input directives

The ASN algorithm described in the present report is activated by a new option `ASN`:

```
OPTI PINS ... ASN
```

From the User's manual: "The so-called "assembled surface normal" (ASN) algorithm of Belytschko and Law (1985) is used to compute a unique (normalized) normal to each external node of the mesh portion subjected to contact, and a unique (normalized) normal to each pinball (parent or descendent). The penetration direction between contacting pinballs is then computed using the ASNs of the two pinballs according to a set of rules. This ameliorates the treatment of flat contact, especially in conjunction with a penalty formulation to compute the contact forces. This option cannot be used together with (is an alternative to) options `FNOR`, `CNOR` (and its sub-options), or `SNOR`."

Visualization directives

New visualization commands in the built-in OpenGL graphical module of the code are provided in order to check the ASN quantities. These can either be activated interactively (right-click and then choose Geometry → Pinballs sub-menu), or in batch mode via the `SCEN` directive:

```
SCEN ... PINB ... PARE CDES ... NORM ... NASN PASN DASN
```

where `PARE` visualizes the parent pinballs, `CDES` the contacting descendent pinballs, `NORM` the contact normals, `NASN` the nodal ASNs (assembled surface normals), `PASN` the pinball ASNs for the parent pinballs, and `DASN` the ASNs for the contacting descendents.

Color code for pinballs

In order to distinguish the various types of pinballs and sub-pinballs the coloring code shown in Table 2 is adopted. Pinballs are represented as glass-like (semi-transparent) spheres of the color listed in the Table.

Pinball type	Status or sub-type	Color
Parent	active	GREEN
	not active	RED
Descendent	Element	RED
	Vertex	MAGENTA
	Corner	CYAN
	Face	BLUE

Table 2 - Color code for the different pinball types.

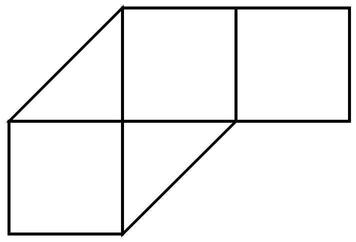


Figure 1 - Continuum elements mesh.

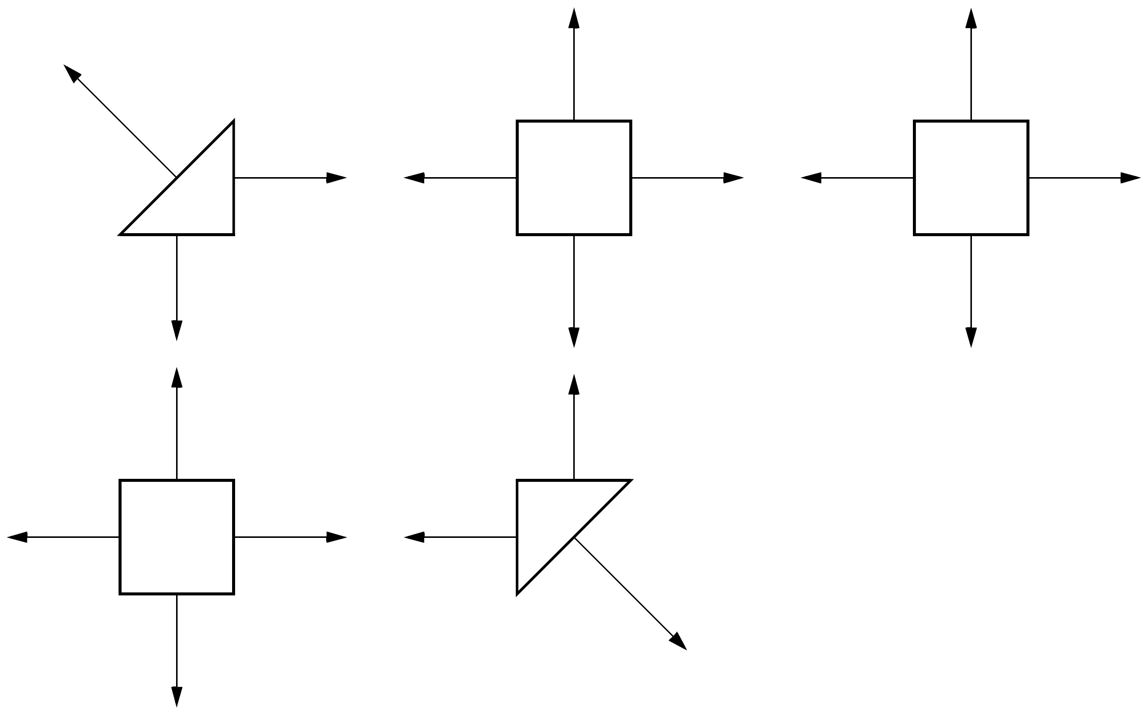


Figure 2 - Element face normals.

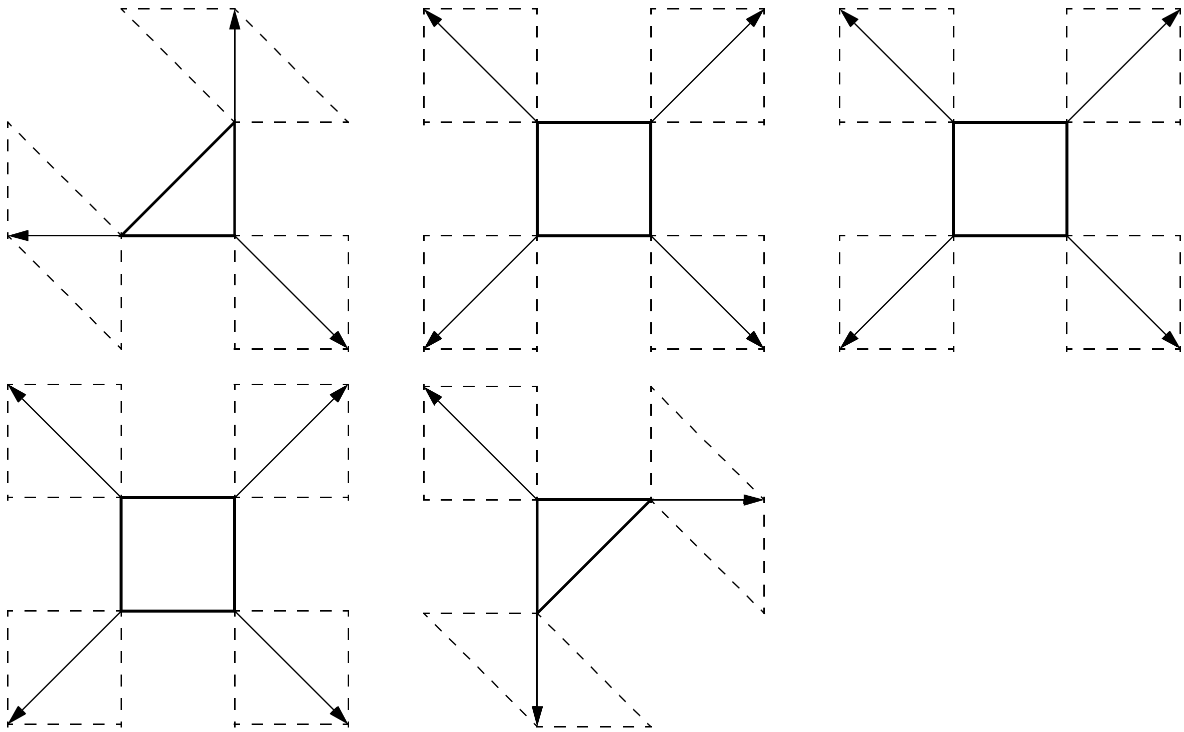


Figure 3 - Assembled element nodal normals.

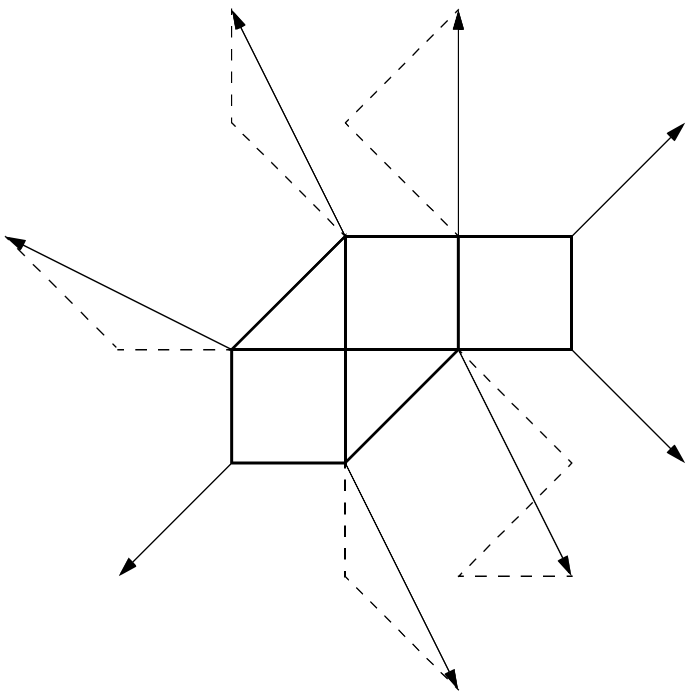


Figure 4 - Assembled global nodal normals.

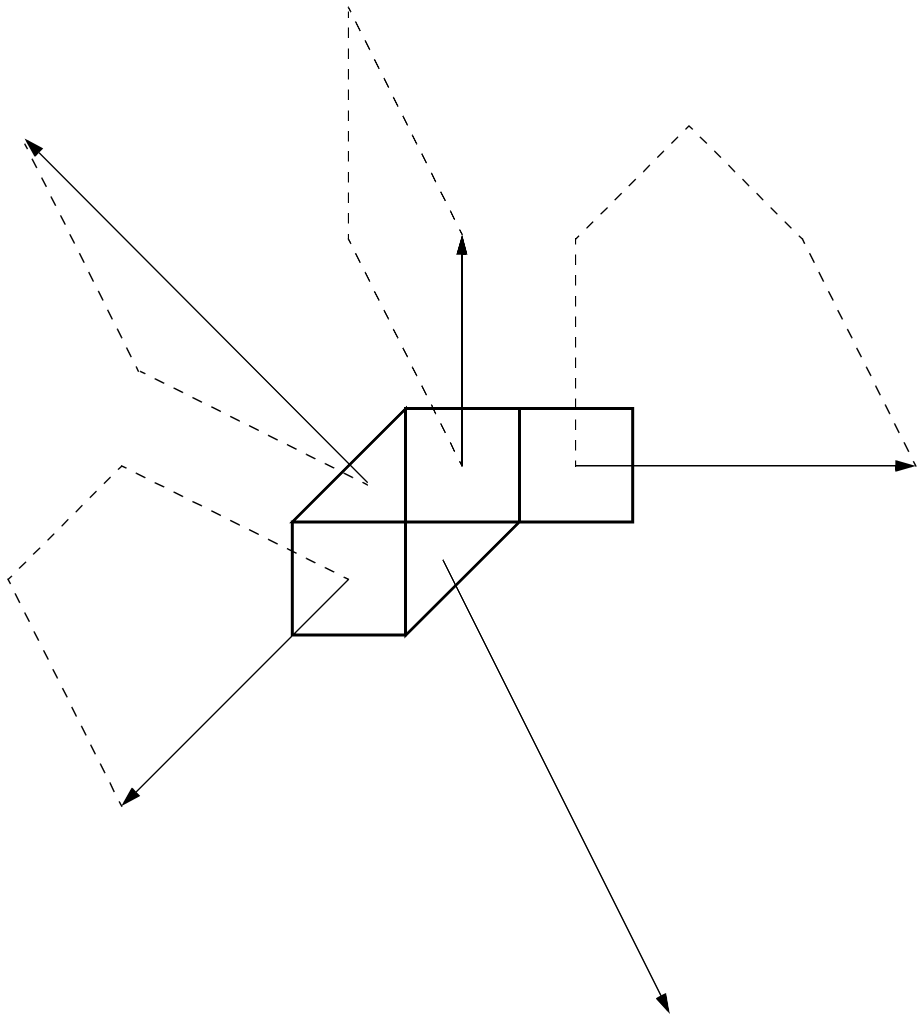


Figure 5 - Assembled global element normals.

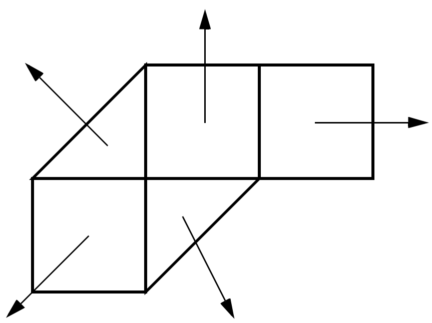


Figure 6 - Normalized element normals.

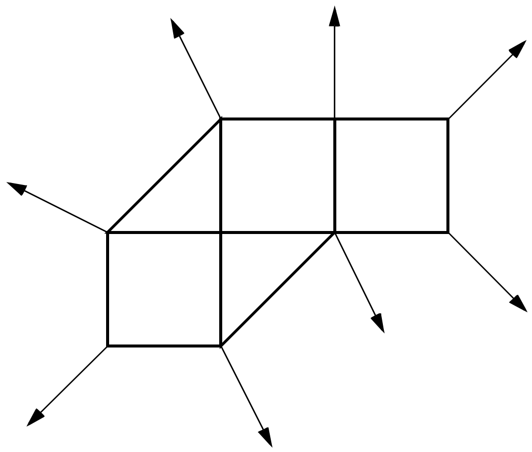


Figure 7 - Normalized global nodal normals.

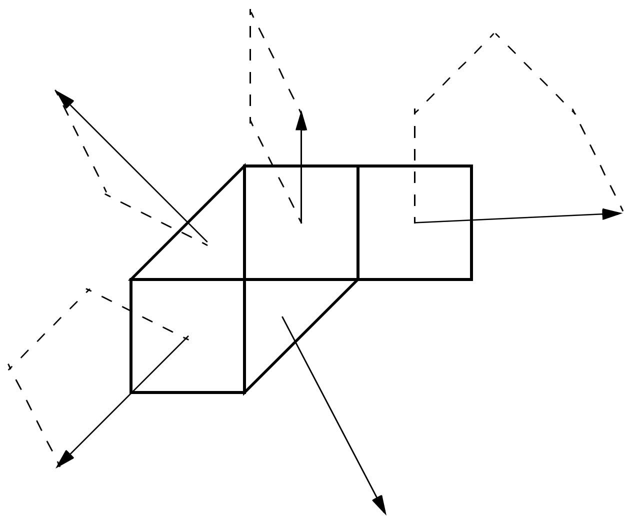


Figure 8 - Assembled global element normals (variant).

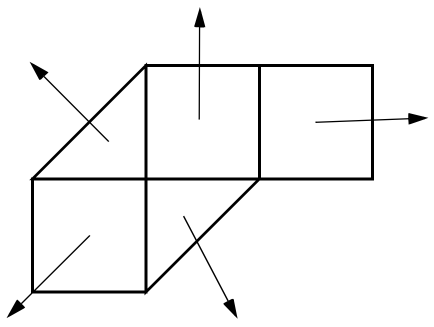


Figure 9 - Normalized global element normals (variant).

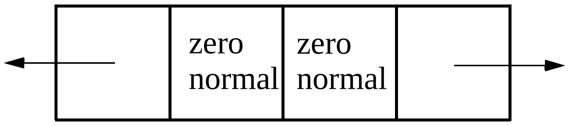


Figure 10 - Pathological case (insufficient discretization).

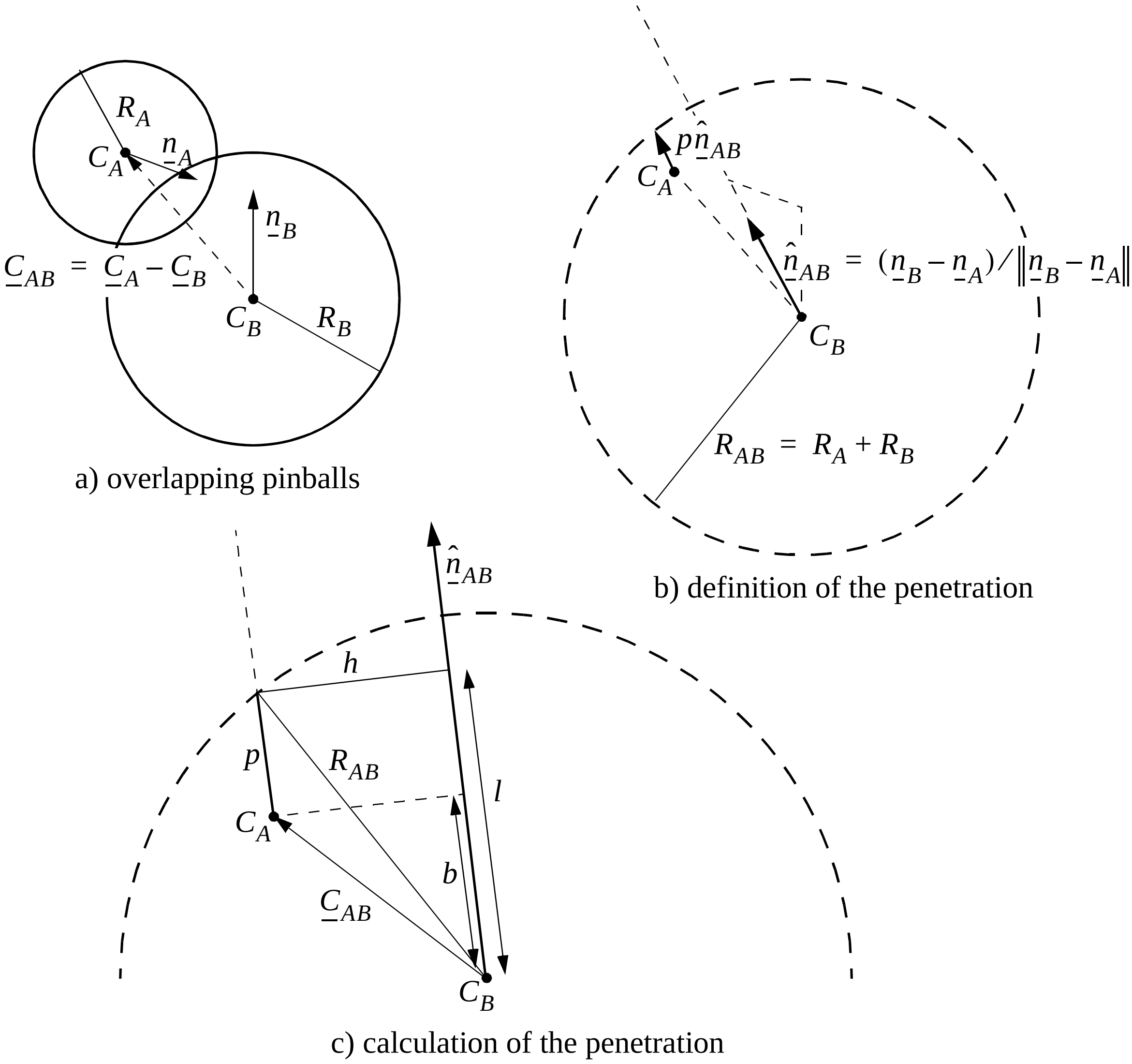


Figure 11 - Penetration between two pinballs.

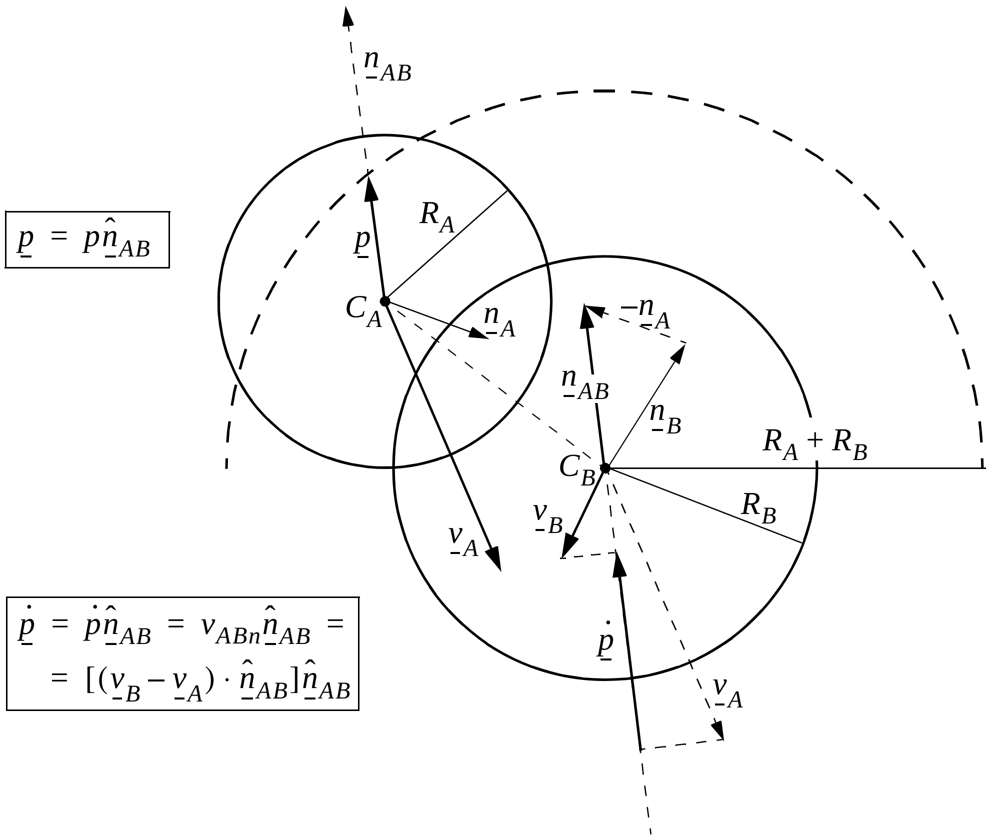


Figure 12 - Rate of penetration between two pinballs.

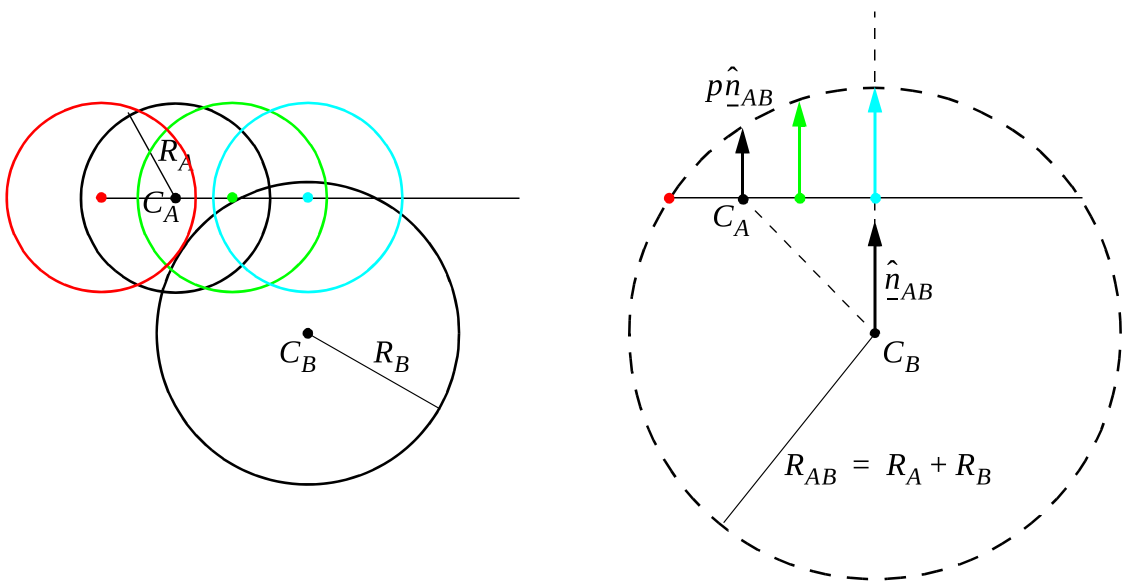


Figure 13 - Variation of the penetration between two pinballs.

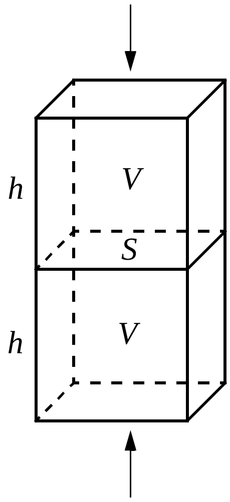


Figure 14 - Contacting hexahedra in 3D.

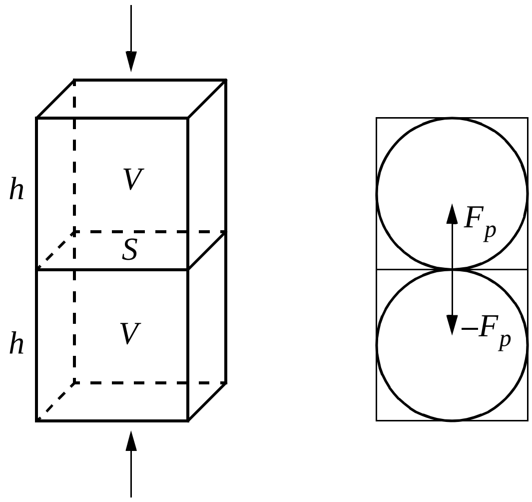


Figure 15 - Contact in 3D computed by zero-level pinballs.

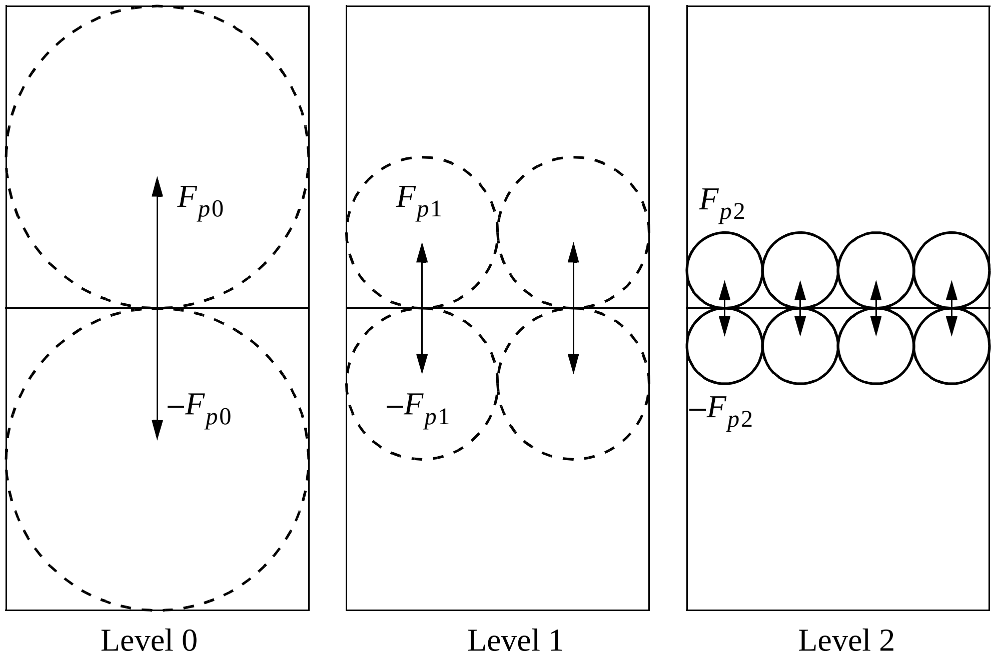


Figure 16 - Contact in 3D computed by hierarchic pinballs.

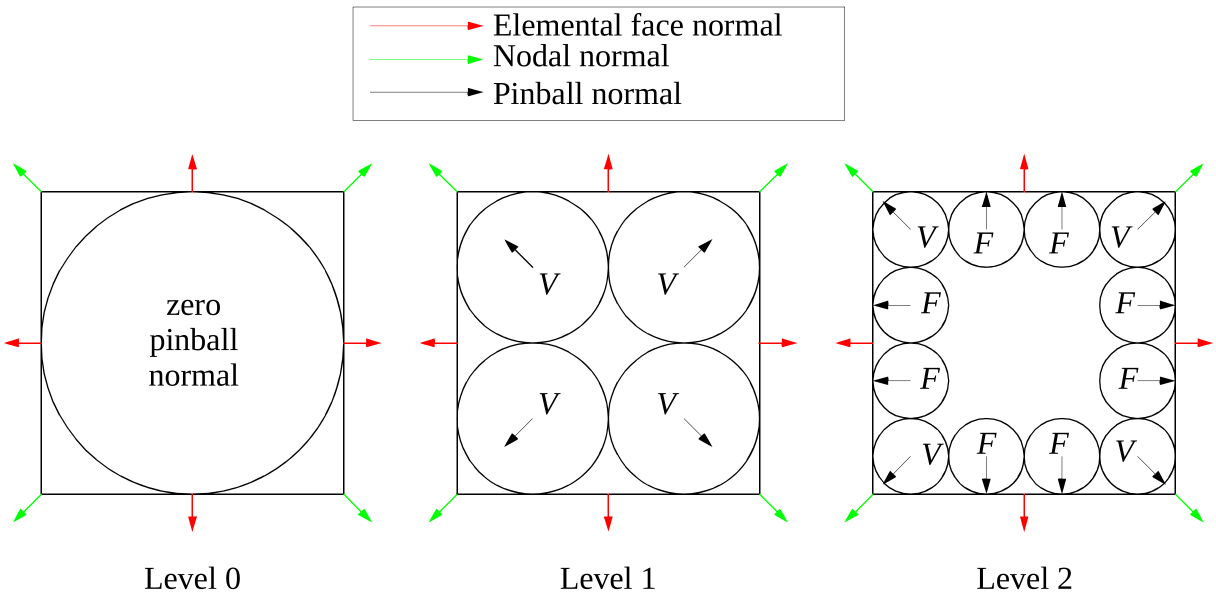


Figure 17 - ASN with hierarchic pinballs on an isolated 2D quadrilateral.

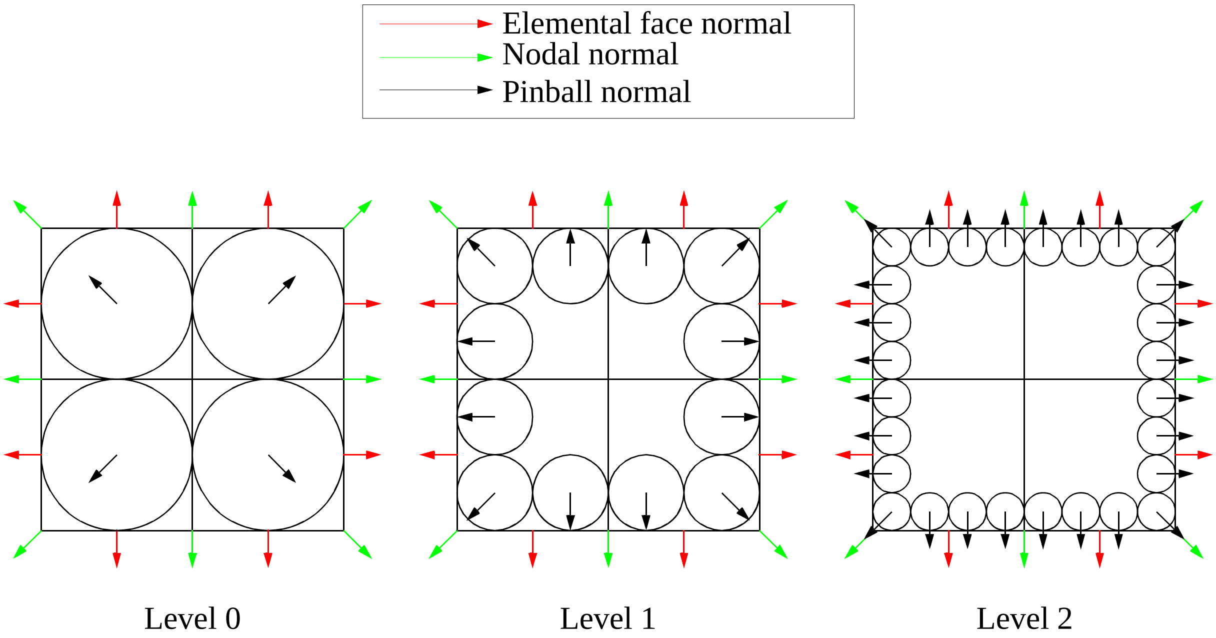


Figure 18 - ASN with hierarchic pinballs on a simple mesh of 2D quadrilaterals.

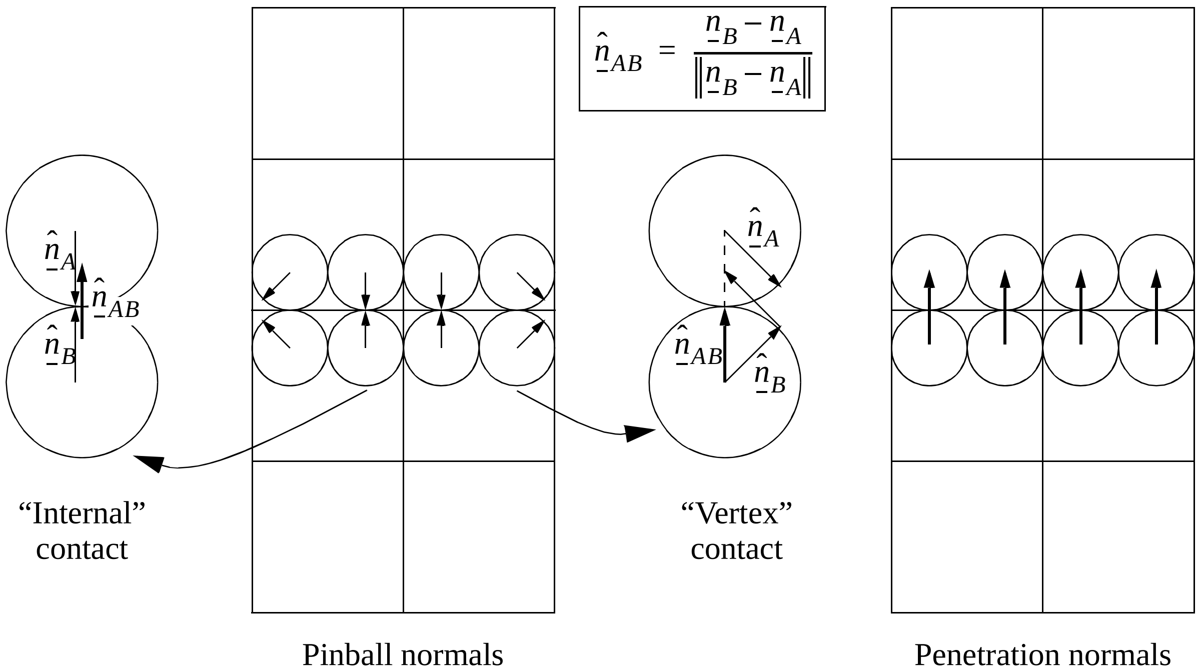


Figure 19 - “Flat” aligned contact with hierarchic pinballs.

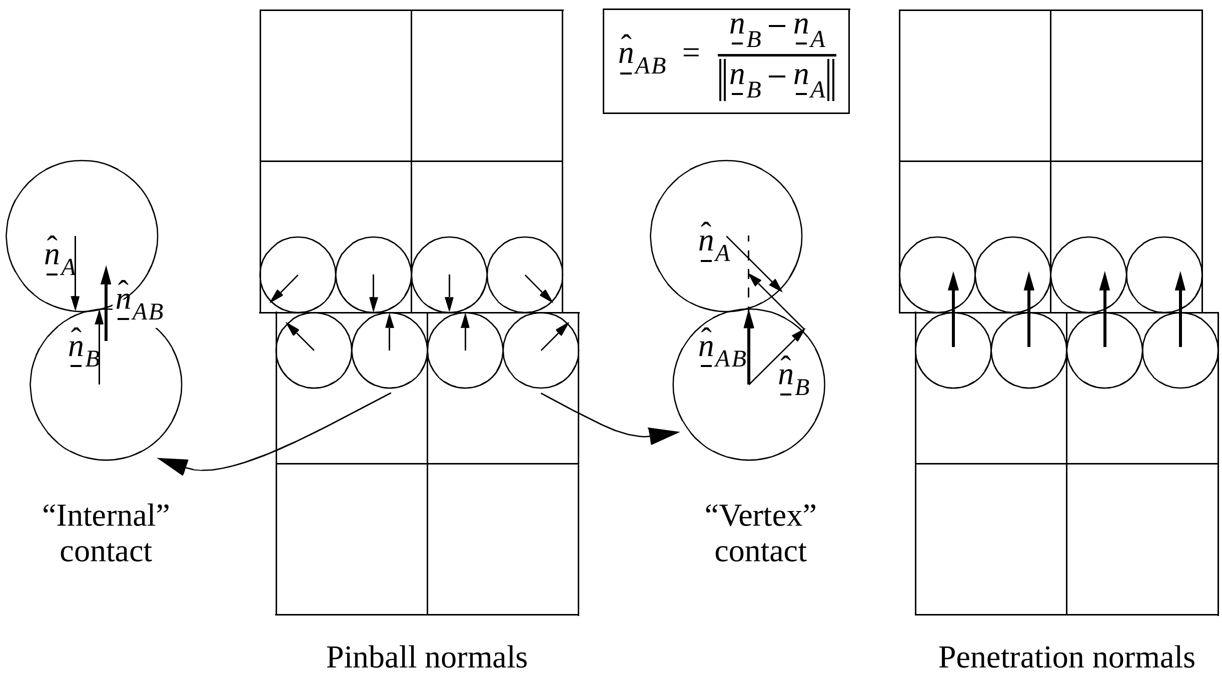


Figure 20 - "Flat" misaligned contact with hierarchic pinballs.

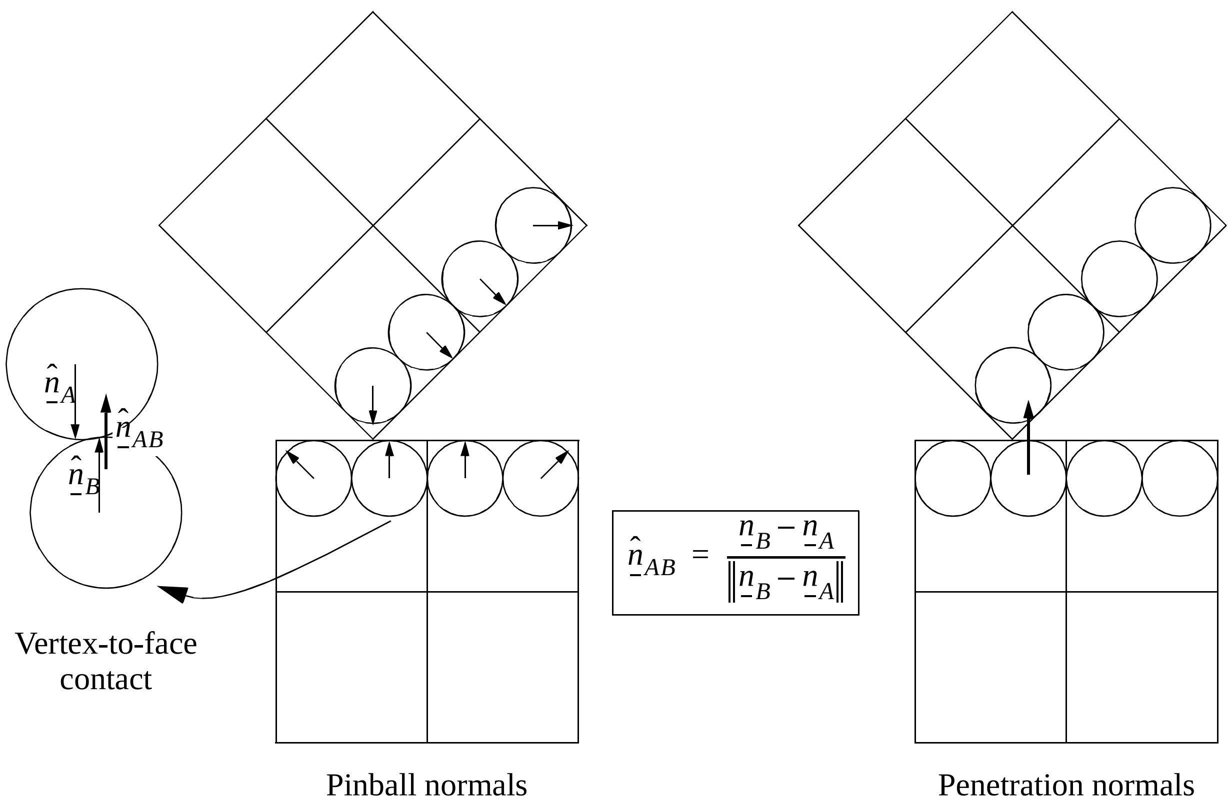


Figure 21 - Vertex-to-face contact with hierarchic pinballs.

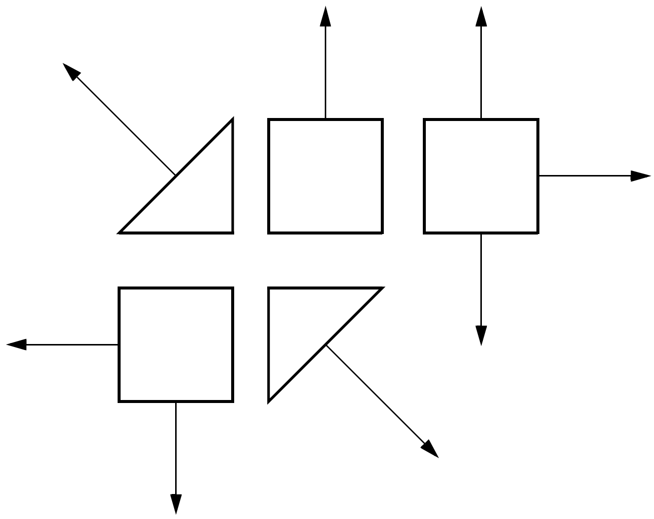


Figure 22 - Element external face normals (built only along external faces).

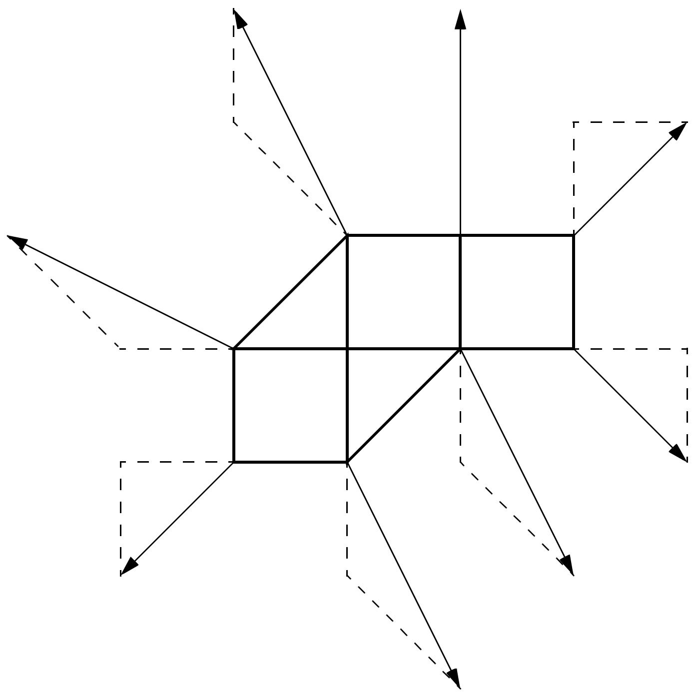


Figure 23 - Assembled global external nodal normals.

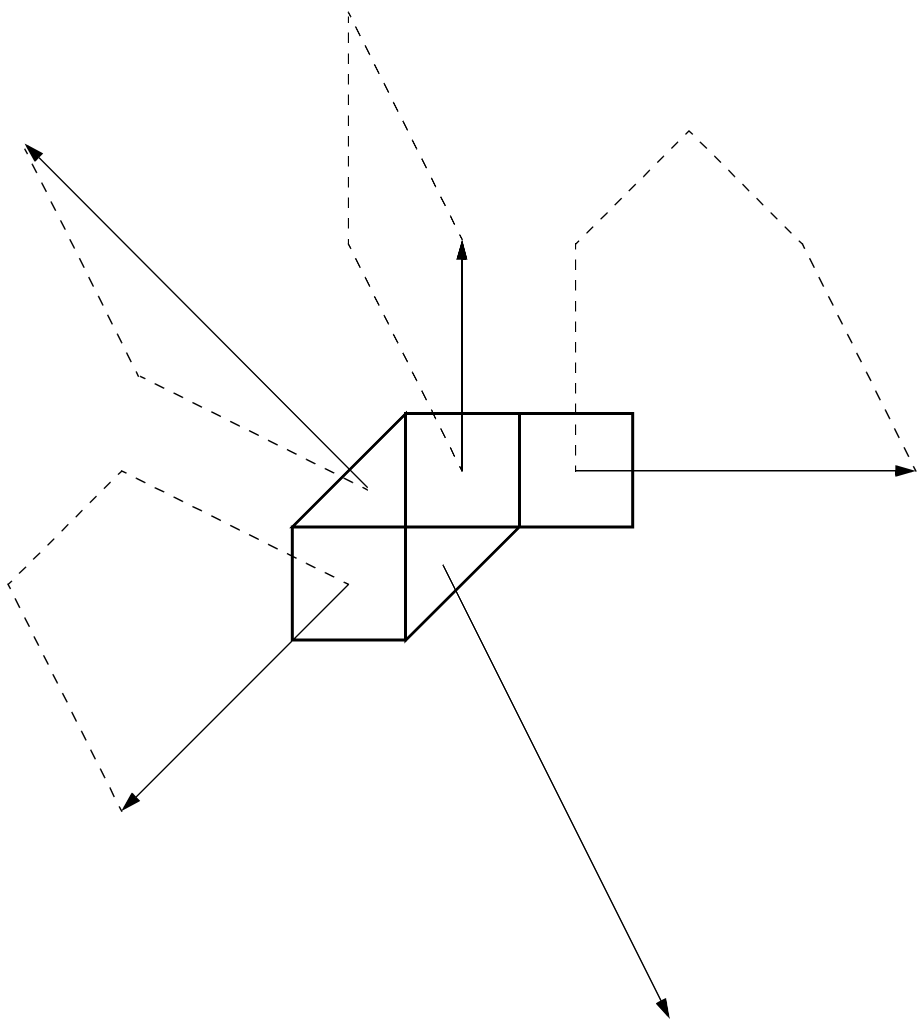


Figure 24 - Assembled global element normals.

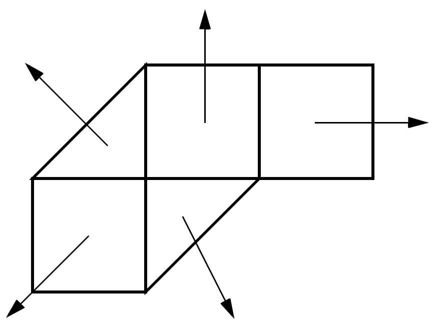


Figure 25 - Normalized element normals.

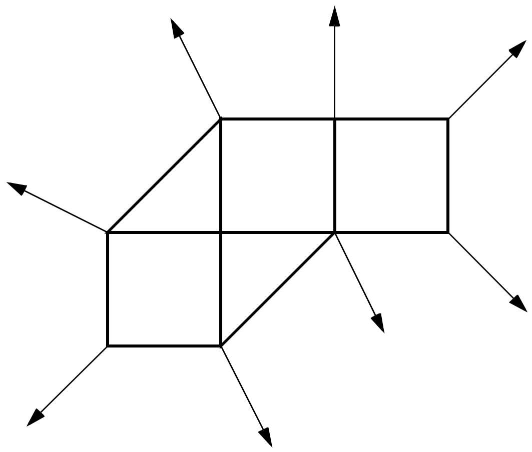


Figure 26 - Normalized global external nodal normals.

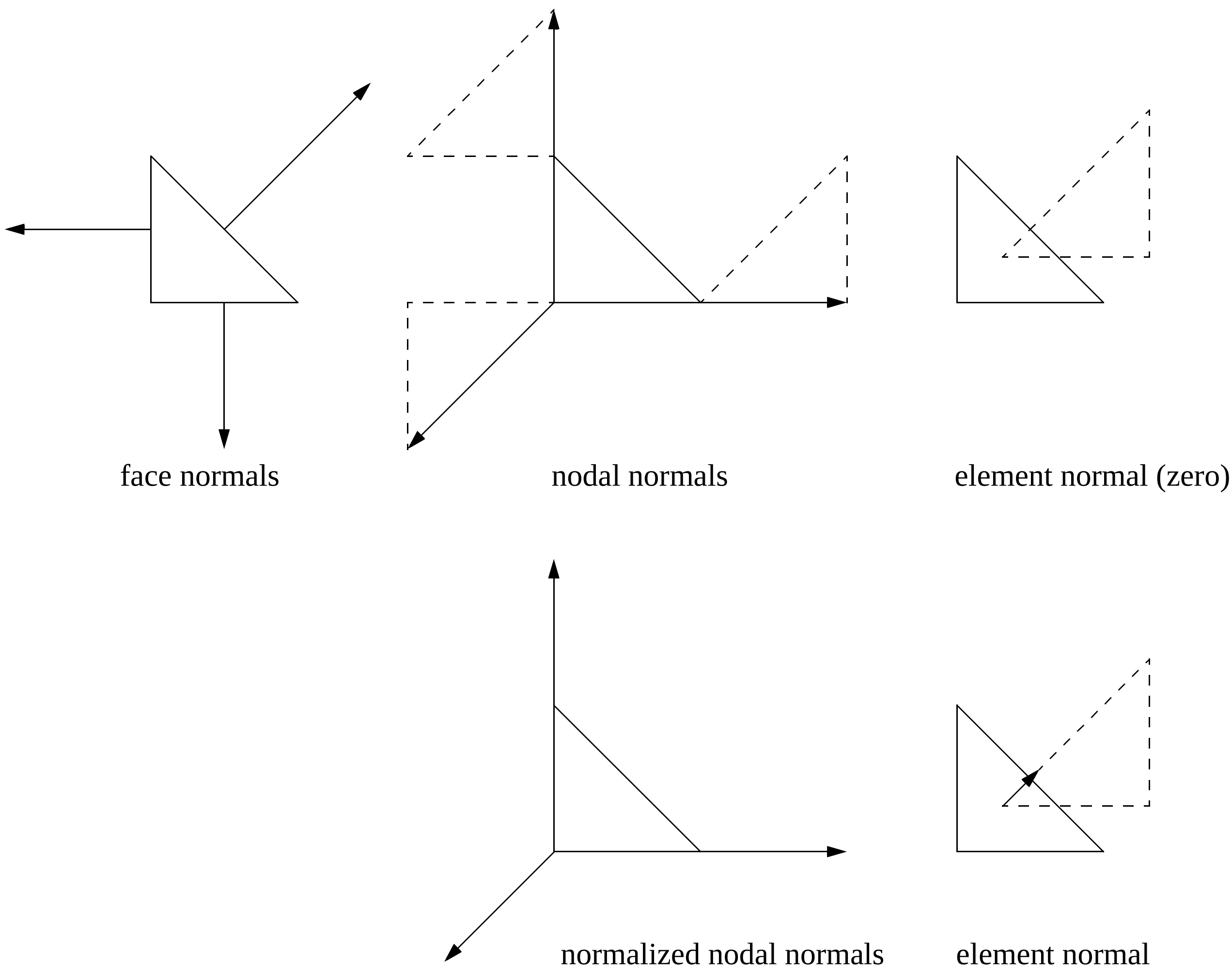


Figure 27 - Example of stand-alone triangle.

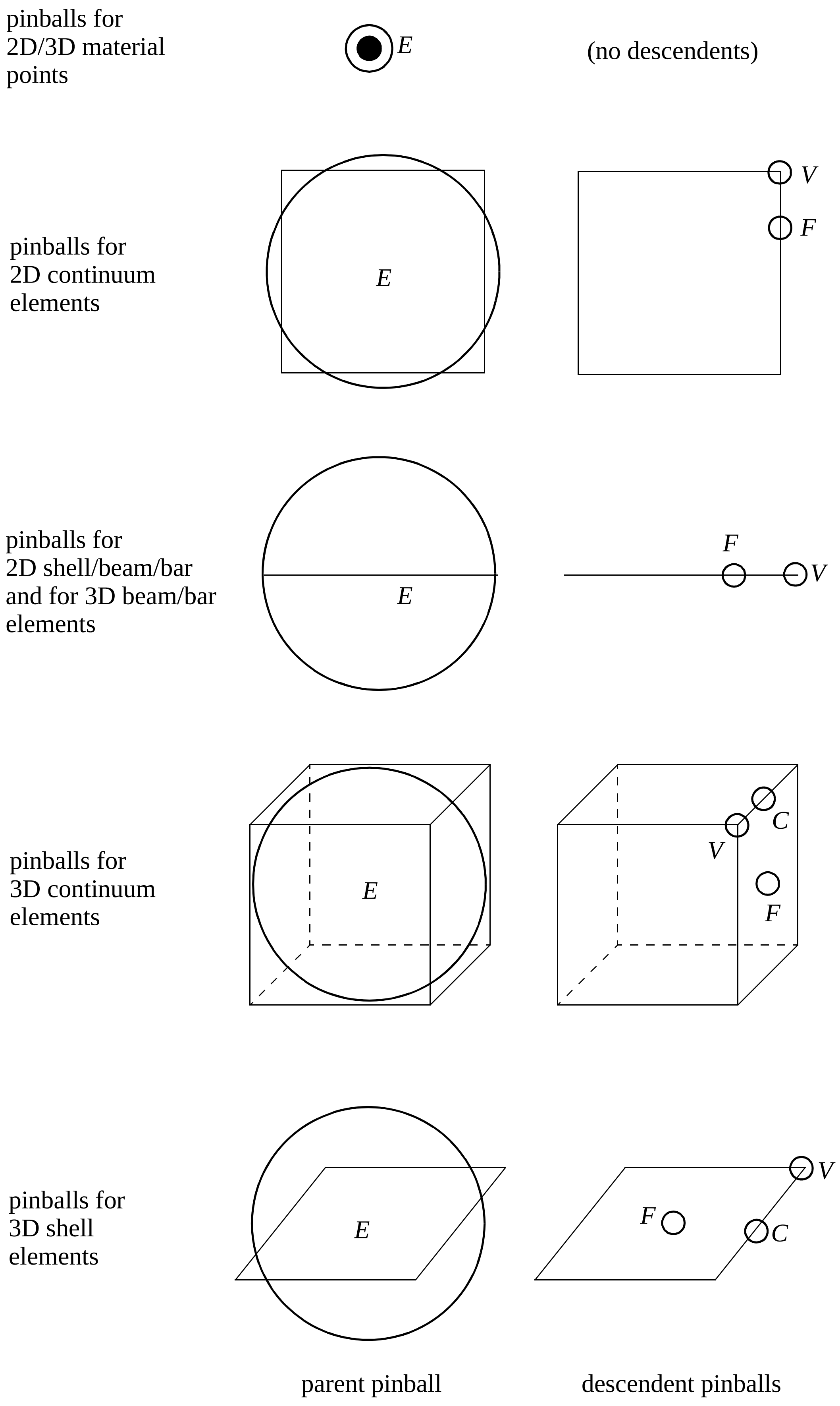


Figure 28 - Pinball types.

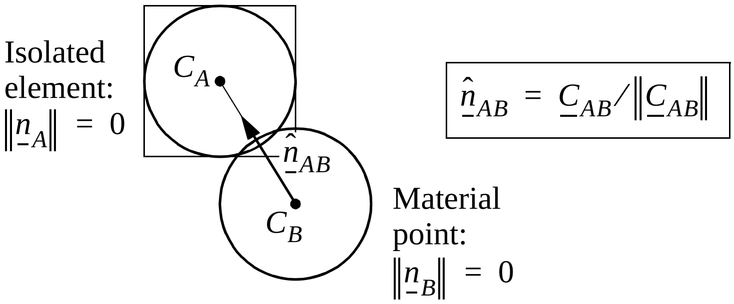


Figure 29 - Penetration direction between two "element" pinballs.

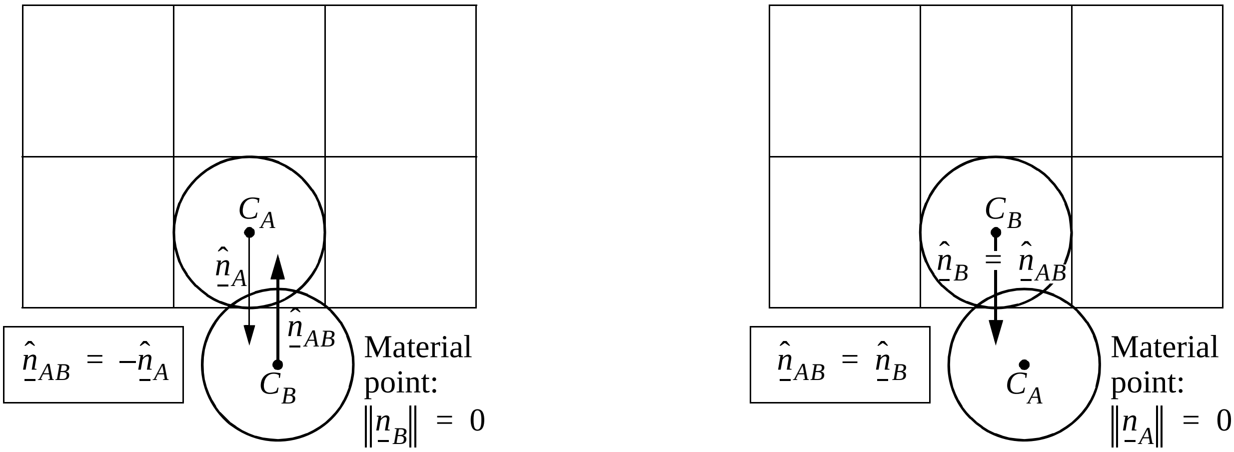


Figure 30 - Penetration direction between two pinballs of which only one has a defined normal.

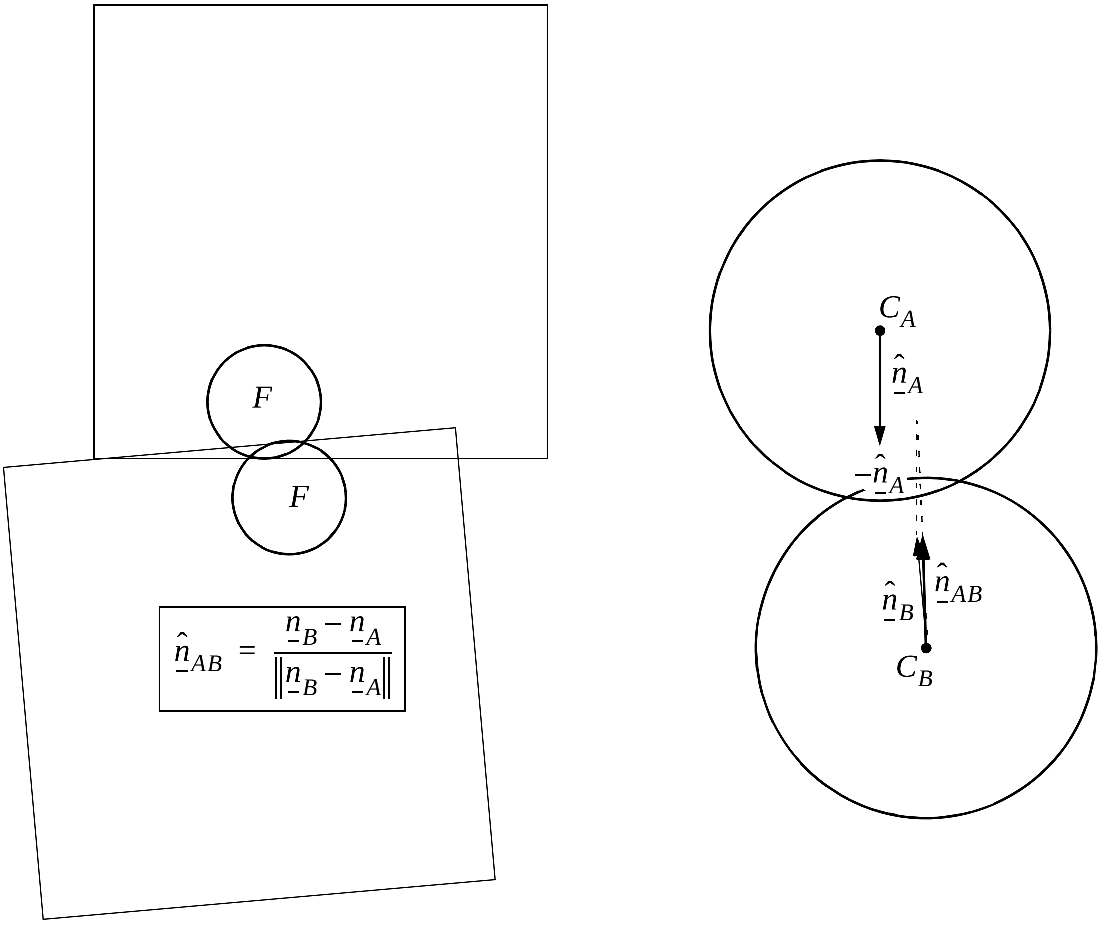


Figure 31 - Penetration direction between two face pinballs.

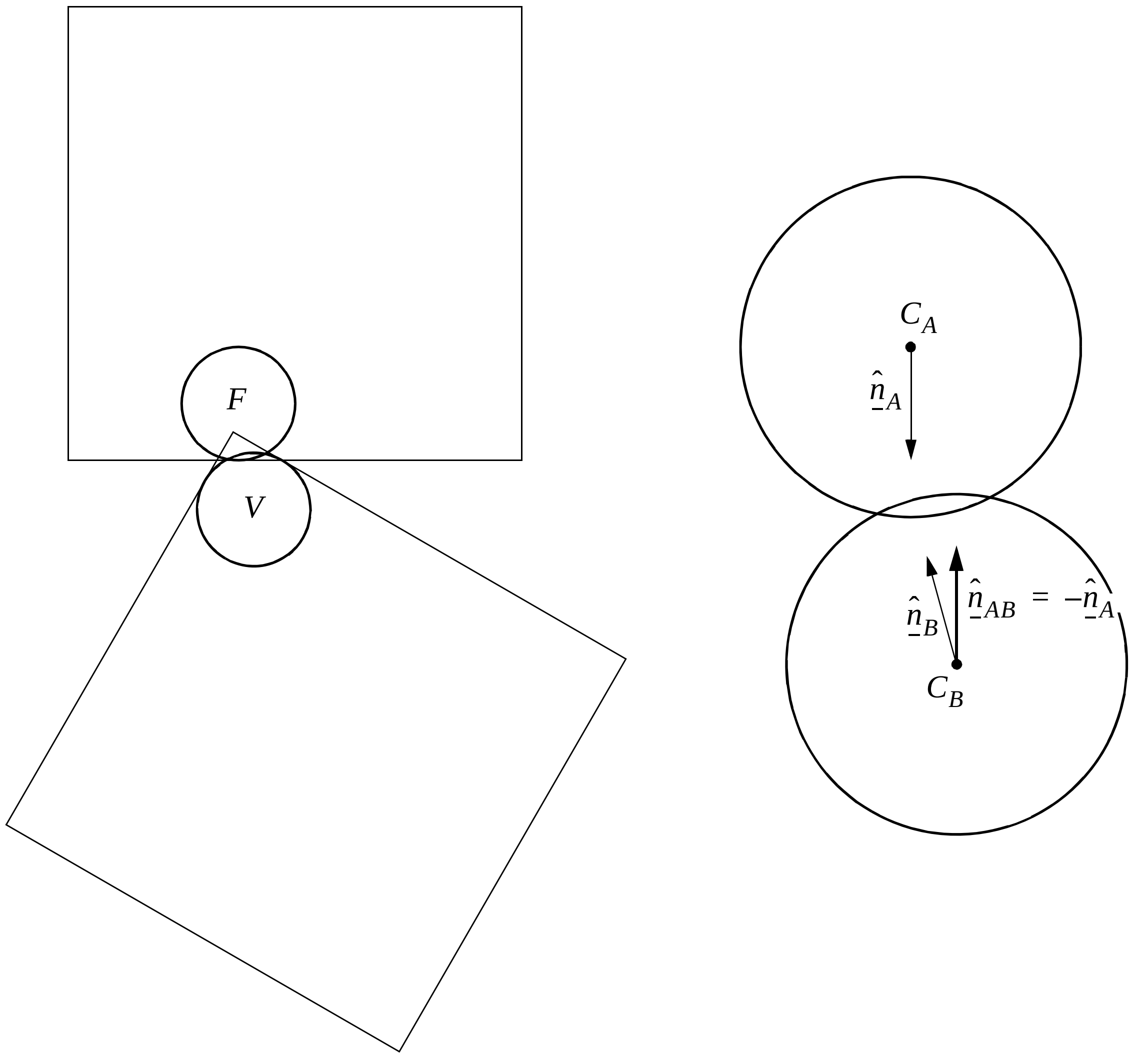


Figure 32 - Penetration direction between a face pinball and a vertex pinball.

3. Contact algorithm for shell/beam/bar elements

We now consider the contacts involving (at least one) non-continuum elements, i.e. a shell, beam or bar element (but not a material point, which can be considered as a special case of continuum element).

3.1 2D case

Let us start from the simplest case, which occurs in 2D. In two spatial dimensions shell, beam and bar elements are all represented by a SEG2 element shape, i.e. a 2-node segment, see Figure 33. The element has two (mutually opposite) faces (with two nodes each) and two edges (with one node each). Other similar elements (in any number, including 0) may be attached to an edge, thus leading in general to element junctions, which are not possible with continuum elements and therefore have not been considered in the previous Chapter. Such elements are called adjacents.

When building element face normals for shell/beam/bar elements, only one of the two faces must be considered. We will consider only the first face, by convention. The second face is equal and opposite to the first one. Of course, unlike the case of continuum (bulky) elements, here the orientation of the normal is known only apart from the sign: one normal direction or the opposite one can be obtained just depending on connectivity, i.e. on the way in which the element nodes are listed in the geometry file.

Therefore, the question arises whether it makes sense or not to build nodal normals for such elements. In Figure 33 we consider the simplest possible case, just two SEG2 elements e_1, e_2 sharing a common node I. Obviously the assembled normal at the common node depends upon the orientation of the element normals, whose sign is arbitrary as mentioned. Therefore, the result would depend upon the specific numbering of the elements, see e.g. cases A and B in the Figure, and this is clearly unacceptable. A similar problem would also occur in the case of junctions, i.e. of three or more elements sharing the same node.

Therefore, it seems preferable to assume that nodal normals cannot be computed (and therefore are simply set at 0.0) for SEG2 elements.

It remains to establish what should be done upon pinball refinement, i.e. at levels $L > 0$. The following strategy is tentatively assumed.

ASNs for pinballs of shell/beam/bar elements in 2D

- At the parent level, shell/beam/bar elements in 2D have one element normal each, which coincides (by convention) with the normal to the first face of the element (the second face is just opposite).

- The nodal normals are not assembled so they remain at 0.0.
- Upon refinement of a shell/beam/bar pinball in 2D, some V (vertex) sub-pinballs and some F (face) sub-pinballs are generated.
- The V pinballs are those “close” to an edge without any adjacents. All other descendent pinballs are F pinballs.
- The V pinballs are assigned a zero ASN, so that in contact calculations the centers-joining line algorithm will be preferably adopted.
- The F pinballs have the same ASN as the corresponding parent pinball (i.e. the normal to the first face of the parent element, which is always non-zero for these elements).

This algorithm is summarized in the example of Figure 34. For generality, a structure with several elements is considered. Nodes I, M and N are edges without any adjacents. Node J is a regular node (at which two elements are joined), while node K is an example of junction (three or more elements meeting at a point).

Note that the V pinballs generated according to the above procedure are descendent pinballs with a zero associated ASN. This is contrary to the rule assumed in the previous Chapter for the case of continuum elements, and the corresponding cases will have to be treated appropriately in the general contact algorithm.

3.2 3D case

In 3D we must distinguish two cases depending upon the element shaped: beams/bars (SEG2 shape), or shells (TRI3 or QUA4 shape).

3.2.1 Beam/bar elements in 3D

The shape of these elements (SEG2) is the same as for their corresponding 2D elements. However, the treatment is slightly different because it is impossible to define a normal to these elements in 3D (while it was possible in 2D). Only the plane containing the normal can be defined: this is the plane normal to the segment representing the element.

ASNs for pinballs of beam/bar elements in 3D

- Beam/bar elements in 3D have zero ASN both at the parent and at the descendent level.
- The nodal normals are not assembled so they remain at 0.0.
- Upon refinement of a beam/bar pinball in 3D, some V (vertex) sub-pinballs and some C (corner) sub-pinballs are generated.

- The V pinballs are those “close” to an edge without any adjacents. All other descendent pinballs are C pinballs.
- Both the V pinballs and the C are assigned a zero ASN.

This algorithm is summarized in the example of Figure 35, which is the equivalent in 3D of Figure 34. For generality, a structure with several elements is considered. Nodes I, M and N are edges without any adjacents. Node J is a regular node (at which two elements are joined), while node K is an example of junction (three or more elements meeting at a point).

Like for 2D SEG2-shaped elements, note that both the V and the C pinballs generated according to the above procedure are descendent pinballs with a zero associated ASN, contrary to the rule assumed in the previous Chapter for the case of continuum elements.

Finally, note that due to the occurrence of C (corner) pinballs with an associated zero ASN, we may be faced with the contact between two such pinballs. Since the ASNs are zero, the general formula (4) cannot be used. In this case, instead of using as penetration direction the line connecting the pinball centers (see Figure 29 and eq. 90) it is more appropriate (and more accurate) to use the line joining the closest points on the two segments, as detailed in Appendix A.

3.2.2 Shell elements in 3D

The shape of shell elements in 3D is either TRI3 (triangle with three nodes) or QUA4 (quadrangle with four nodes). These elements have two (mutually opposite) faces with three or four nodes, respectively, and either three or four edges (respectively), each having the shape of a segment with two nodes.

A normal to the element is defined as the normal to the first face, by convention. For the triangle, this normal is geometrically well defined (apart from the sign of course) since the triangle lies in a plane. Instead, the quadrilateral may be warped so that its four nodes do not lie in a common plane. The normal is taken by convention as the perpendicular to the plane defined by the two medians of the quadrilateral: these two lines intersect in a point, the element centroid, and therefore they always lie on a plane.

The calculation of ASNs for the triangular and quadrilateral shells (considered as stand-alone elements) is shown in Figures 36 and 37, respectively, and is detailed hereafter. The case of two elements connected along an edge is shown in Figure 38.

ASNs for pinballs of shell elements in 3D

- At the parent level, pinballs are E pinballs. The associated ASN is taken as the normal to the first face of the element, by convention.

- Nodal ASNs are not assembled so they remain at zero.
- Upon refinement of a shell element in 3D, some F (face) sub-pinballs, some C (corner) sub-pinballs and some V (vertex) sub-pinballs are generated.
- The V pinballs are close to the element vertices. They are assigned zero ASNs.
- The C pinballs are near the element corners without any adjacents. They are also assigned zero ASNs.
- The F pinballs are either in the interior of the element or near the element corners with adjacents. (see e.g. Figure 38). They have the same ASN as the parent pinball, i.e. the normal to the (first) element face.

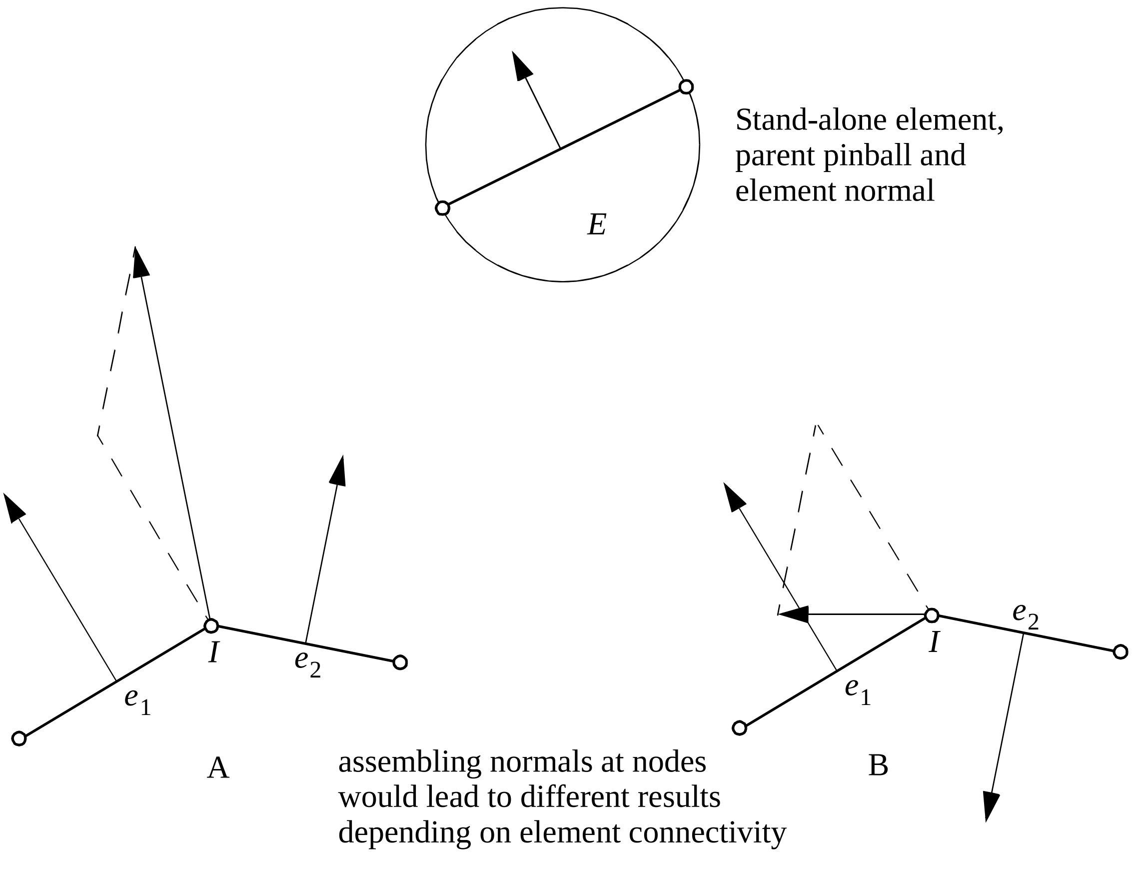
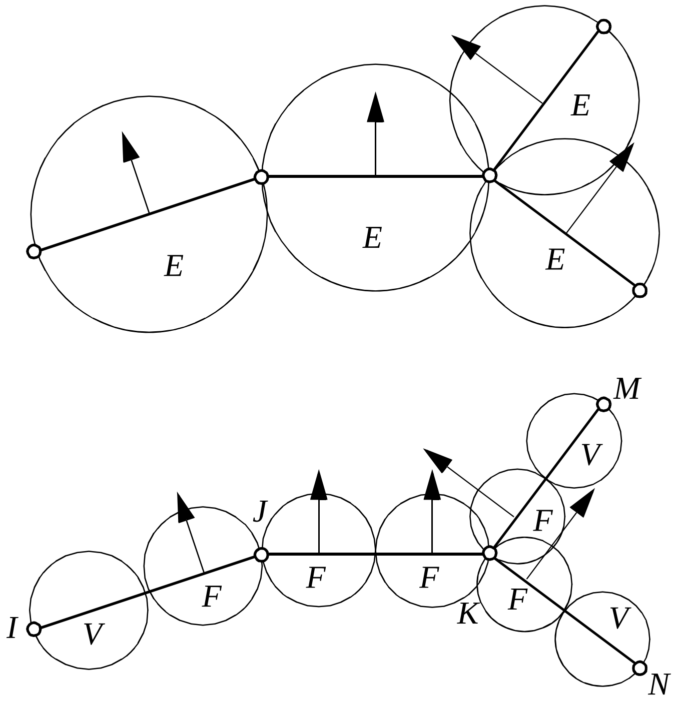


Figure 33 - A shell/beam/bar element in 2D (SEG2 element shape).



$L = 0$

Parent pinballs are all *E* pinballs

Element ASNs are normal to 1st face of the element

Nodal ASNs are zero

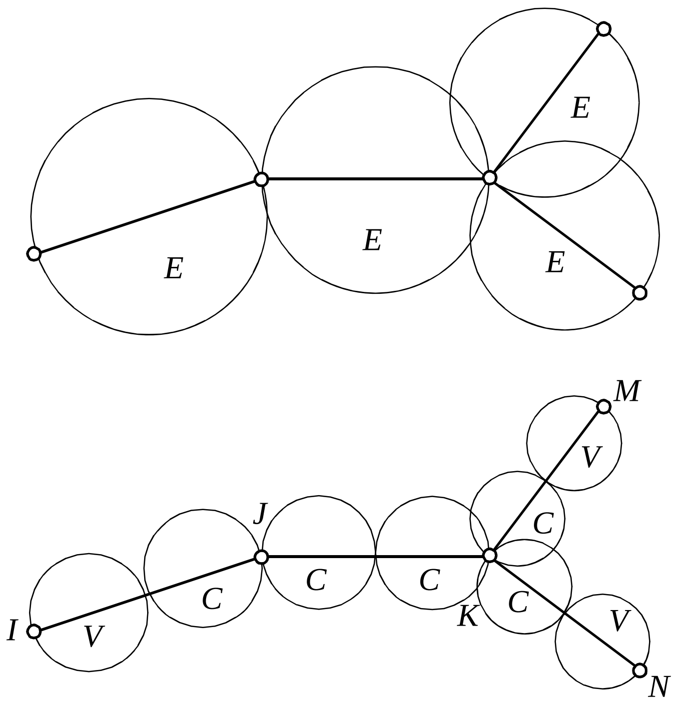
$L > 0$

Descendent pinballs are either *V* or *F* pinballs

V pinballs are near the edges without adjacents and have zero ASNs

E pinballs have same ASN as parent

Figure 34 - ASNs for shell/beam/bar elements in 2D (SEG2 element shape).



$L = 0$

Parent pinballs are all *E* pinballs

Element ASNs are zero

Nodal ASNs are zero

$L > 0$

Descendent pinballs are either *V* or *C* pinballs

V pinballs are near the edges without adjacents and have zero ASNs

C pinballs also have zero ASN

Figure 35 - ASNs for beam/bar elements in 3D (SEG2 element shape).

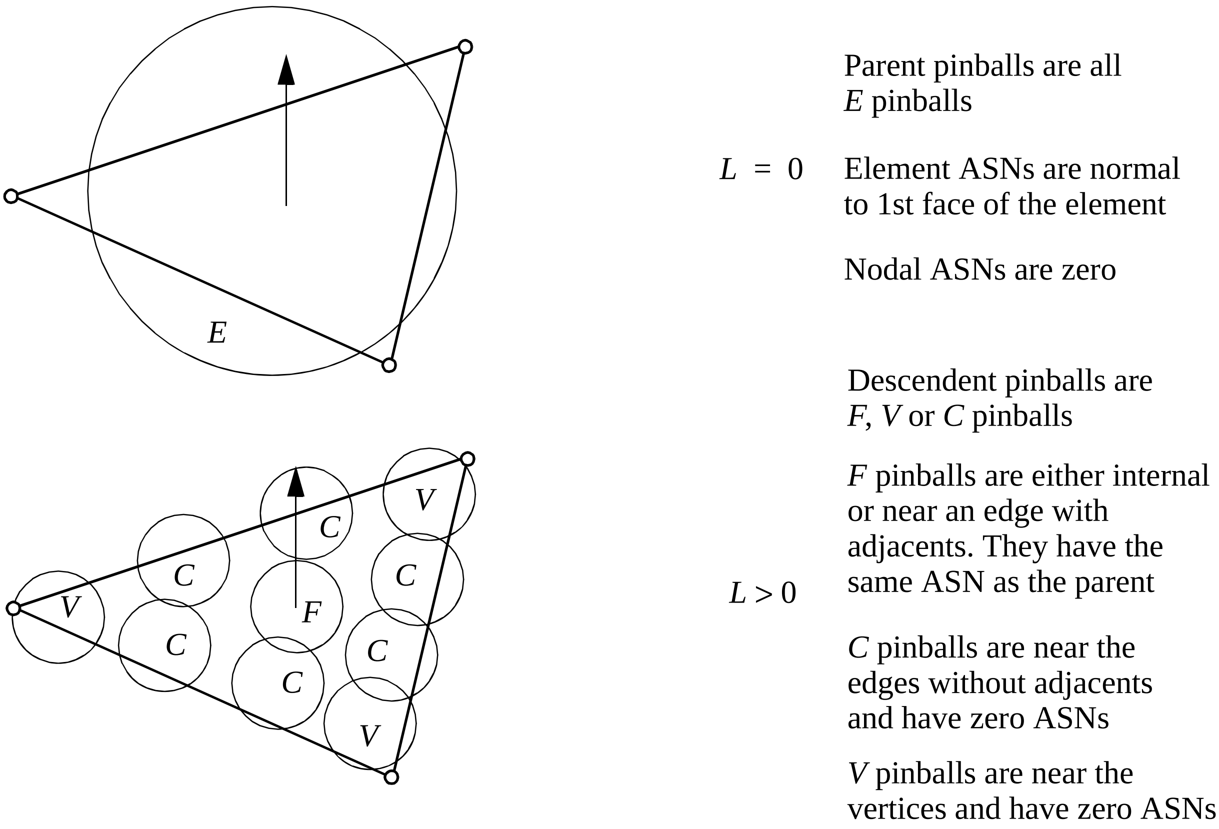


Figure 36 - ASNs for a stand-alone shell element in 3D (TRI3 element shape).

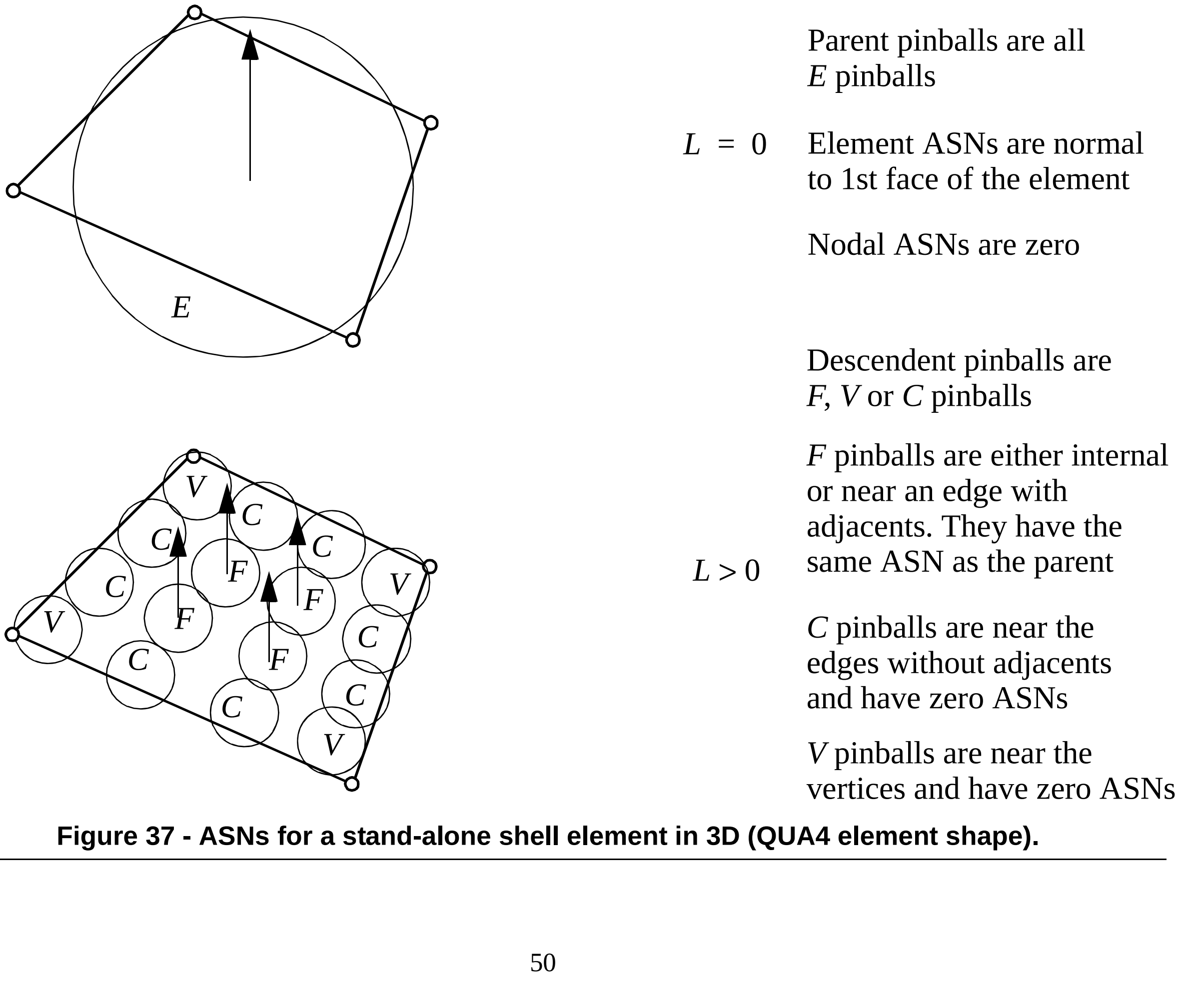


Figure 37 - ASNs for a stand-alone shell element in 3D (QUA4 element shape).

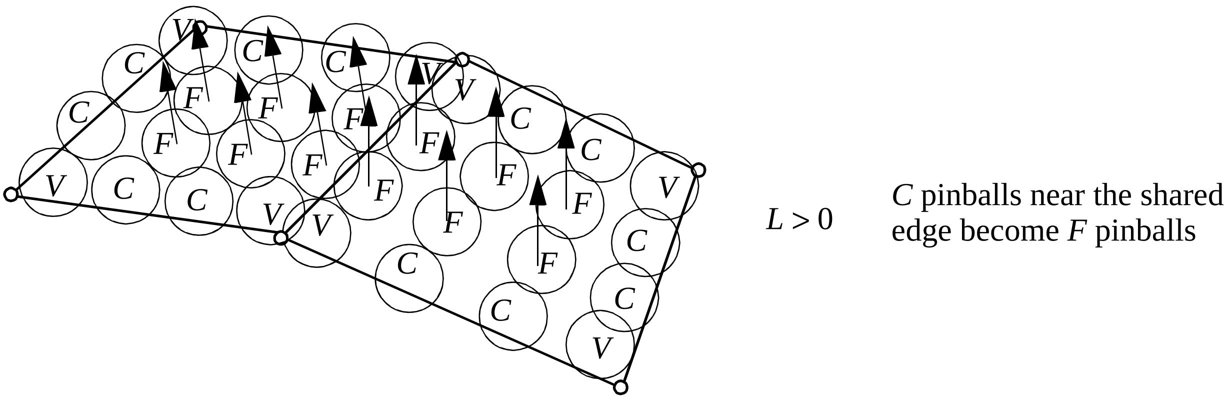


Figure 38 - ASNs for two shell element in 3D sharing a common edge.

4. Implementation details

Some details of the implementation are presented hereafter for completeness.

In the module M_PINBALL, in the derived type `TYPE PINBALL` (which represents parent pinballs) the following description of the ASN is added:

```
TYPE PINBALL ! "PARENT" PINBALL DEFINITION
  INTEGER :: ELEMENT      ! ELEMENT INDEX
  REAL(8) :: RADIUS       ! RADIUS OF THE PINBALL
  REAL(8) :: CENTER(3)    ! CENTER OF THE PINBALL
  INTEGER :: MAXLEV       ! MAXIMUM LEVEL OF DESCENDENT PINBALLS
  INTEGER :: SET          ! SET TO WHICH PINBALL BELONGS
                          ! (<0 IF SELF-CONTACT ENABLED)
  INTEGER :: HARDNESS     ! ASSOCIATED "HARDNESS" (ONLY FOR FLAT
                          ! CONTACT)
  REAL(8) :: DTPINB       ! LIMIT TIME STEP ON THIS PINBALL'S
                          ! ELEMENT AS DICTATED BY PCONTACTS
                          ! (0 IF NONE)
  LOGICAL :: IS_ACTIVE    ! ACTIVE PINBALL OR NOT
  REAL(8) :: ASN(4)       ! ASN OF THE PINBALL:
                          ! (1)   = LENGTH OF THE ASN BEFORE
                          !         NORMALIZATION (>= 0.0)
                          ! (2:) = COMPONENTS OF THE NORMALIZED
                          !         ASN (IDIM VALUES) OF LENGTH 1.0
                          !         (IF ASN(1)>0) OR 0.0 (IF ASN(1)=0)
END TYPE PINBALL
```

Similarly, in the derived type `TYPE DESCENDENT_PINBALL` (which represents descendent pinballs) we add the same quantity:

```
TYPE DESCENDENT_PINBALL ! "DESCENDENT" PINBALL DEFINITION
  INTEGER :: ANCESTOR     ! INDEX OF ANCESTOR (0-LEVEL) PINBALL
  INTEGER :: LEVEL        ! LEVEL OF THIS PINBALL (>= 0)
  REAL(8), POINTER :: XYZ(:,:) ! COORDS OF "NODES" OF PINBALL
  REAL(8) :: RADIUS       ! RADIUS OF THE PINBALL
  REAL(8) :: CENTER(3)    ! CENTER OF THE PINBALL
  INTEGER, POINTER :: IFACE(:) ! FACE INDEXES
                               !   FOR CONTINUUM ELEMENTS:
                               ! N>0=EXTERNAL, ON FACE N OF PARENT
                               ! M<0=INTERNAL, SEES PAR. NEIGHBOUR M
                               !   0=INTERNAL (FROM CUT OF PARENT)
                               !   FOR BEAM/SHELL ELEMENTS:
                               ! SPECIAL CONVENTION (SEE REPORT)
  REAL(8) :: ASN(4)       ! ASN OF THE DESCENDENT PINBALL:
                          ! (1)   = LENGTH OF THE ASN BEFORE
                          !         NORMALIZATION (ALWAYS > 0.0)
                          ! (2:) = COMPONENTS OF THE NORMALIZED
                          !         ASN (IDIM VALUES) OF LENGTH 1.0
END TYPE DESCENDENT_PINBALL
```

In the same module M_PINBALL, a new possible value for the `PINBALL_FNOR` variable is added: the variable is set to 3 when the ASN optional keyword is read (`OPTI PINS ASN`):

```
INTEGER :: PINBALL_FNOR  ! 0 : PINBALL VELOCITY CONSTRAINT IS
                         !     WRITTEN ALONG THE DIRECTION OF
                         !     THE LINE THAT JOINS THE CENTERS
                         !     (DEFAULT)
                         ! 1 : PINBALL VELOCITY CONSTRAINT IS
                         !     WRITTEN ALONG A "MEAN" OF THE
                         !     TWO FACE NORMALS N = (NA - NB)
                         !     (OPTI PINS FNOR)
                         !     (THIS REQUIRES OPTI PINS FACE
```

```
!      OR OPTI PINS FACI)
! 2 : PINBALL VELOCITY CONSTRAINT IS
!      WRITTEN ALONG A "COMMON" NORMAL
!      TO ALL CONTACTING SUB_PINBALL
!      COUPLES (OPTI PINS CNOR)
! 3 : PINBALLS INTERACTION OCCURS
!      ALONG THE DIRECTION RESULTING FROM
!      THE ASSEMBLED SURFACE NORMALS
!      (ASN) ASSOCIATED WITH EACH
!      PINBALL (PARENT OR DESCENDENT)
!      (OPTI PINS ASN)
```

In module M_PINBALLS_DATA we add a new array `NODAL_ASN(:,:)` to contain the assembled normals at nodes:

```
      REAL(8), POINTER :: NODAL_ASN(:,:)
*
* for use with opti pinb asn:
* nodal_asn(1,i)   = norm of nodal asn at node i (>= 0.0)
* nodal_asn(2:,i)  = components of nodal asn at node i (idim values)
*                    normalized to length 1.0 or 0.0
```

If needed, the ASNs for pinballs and at nodes are computed in the subroutine M_PINBALLS:: UPDATE_PINBALLS. The penalty contact forces are computed in M_PINBALLS:: PENA_CONTACT_FORCES.

The contact normal is computed in M_PINBALLS_SPLIT::CONTACT_DATA_PARENTS in the case of contact between parent (0-level) pinballs, or in M_PINBALLS_SPLIT::CONTACT_DATA in the case of contact between descendent pinballs.

The pinball types for each couple of contacting pinballs (`TYPEI`, `TYPEJ`) are added to the derived type `TYPE PCONTACT` in module M_PCONTACT:

```
      TYPE PCONTACT
        INTEGER :: PA, PB       ! INDEXES OF THE TWO (PARENT) PINBALLS
        REAL(8) :: NAB(3)       ! UNIT NORMAL FROM A TO B
                                ! "JOINS CENTERS"    IF PINBALL_FNOR = 0,
                                ! "MEAN" NORMAL      IF PINBALL_FNOR = 1,
                                ! "COMMON" NORMAL    IF PINBALL_FNOR = 2,
                                ! "ASN" BUILT NORMAL IF PINBALL_FNOR = 3
        REAL(8), POINTER :: SHA(:), SHB(:)
                                ! SHAPE FUNCTIONS OF ELEMENT A AND B AT
                                ! INTERACTION POINTS A' AND B'
        REAL(8) :: CSIA(3), CSIB(3) ! NORMALIZED COORS OF CONTACT POINTS
                                ! WITH RESPECT TO PARENT ELEMENTS
        TYPE(DESCENDENT_PINBALL) :: DPI, DPJ ! DESCENDENT PINBALLS IN
                                ! CONTACT (WITH LEVEL=0 IF PARENT)
                                ! THESE ARE COPIES (NOT POINTERS)
        LOGICAL :: HAS_COMMON_NORMAL ! IS ASSOCIATED NORMAL "COMMON"?
        INTEGER :: NODE1, NODE2 ! NODES OF THE CONTACT (EACH MAY BE 0)
                                 ! N1 N2  --> NN CONSTRAINT
                                 !  0  0  --> PP CONSTRAINT
                                 ! N1  0  OR
                                 !  0 N2  --> NP CONSTRAINT
        INTEGER :: ELEM1, ELEM2 ! ELEMENTS OF THE CONTACT (EACH MAY BE 0)
                                 !  0  0  --> NN CONSTRAINT
                                 ! E1 E2  --> PP CONSTRAINT
                                 !  0 E2  OR
                                 ! E1  0  --> NP CONSTRAINT
        LOGICAL :: FLAG          ! GENERIC FLAG FOR VARIOUS OPERATIONS
        REAL(8) :: PENETR        ! PENETRATION (ALONG NAB) IF USEFUL
```

```
   REAL(8) :: PDOT          ! PENETRATION RATE (ALONG NAB) IF USEFUL
   INTEGER :: TYPEI, TYPEJ ! TYPE OF CONTACTING (SUB-)PINBALL:
                            ! 0 = ELEMENT (PARENT) PINBALL
                            ! 1 = VERTEX SUB-PINBALL
                            ! 2 = CORNER SUB-PINBALL (3D ONLY)
                            ! 3 = FACE SUB-PINBALL
   INTEGER :: NODEI(5), NODEJ(5) ! N. OF CONTACTING SUB-PINBALL NODES,
                            ! THEN LIST OF THESE NODES (GLOBAL INDEXES):
                            ! 1 NODE FOR A VERTEX SUB-PINBALL
                            ! 2 NODES FOR A CORNER SUB-PINBALL (3D ONLY)
                            ! 2, 3 OR 4 NODES FOR A FACE SUB-PINBALL
END TYPE PCONTACT
```

In addition, the nodes `NODEI(:)`, `NODEJ(:)` related to the contacting sub-pinballs are also added, see above list of the derived type.

A flowchart of the relevant code parts is shown in Figure 39.

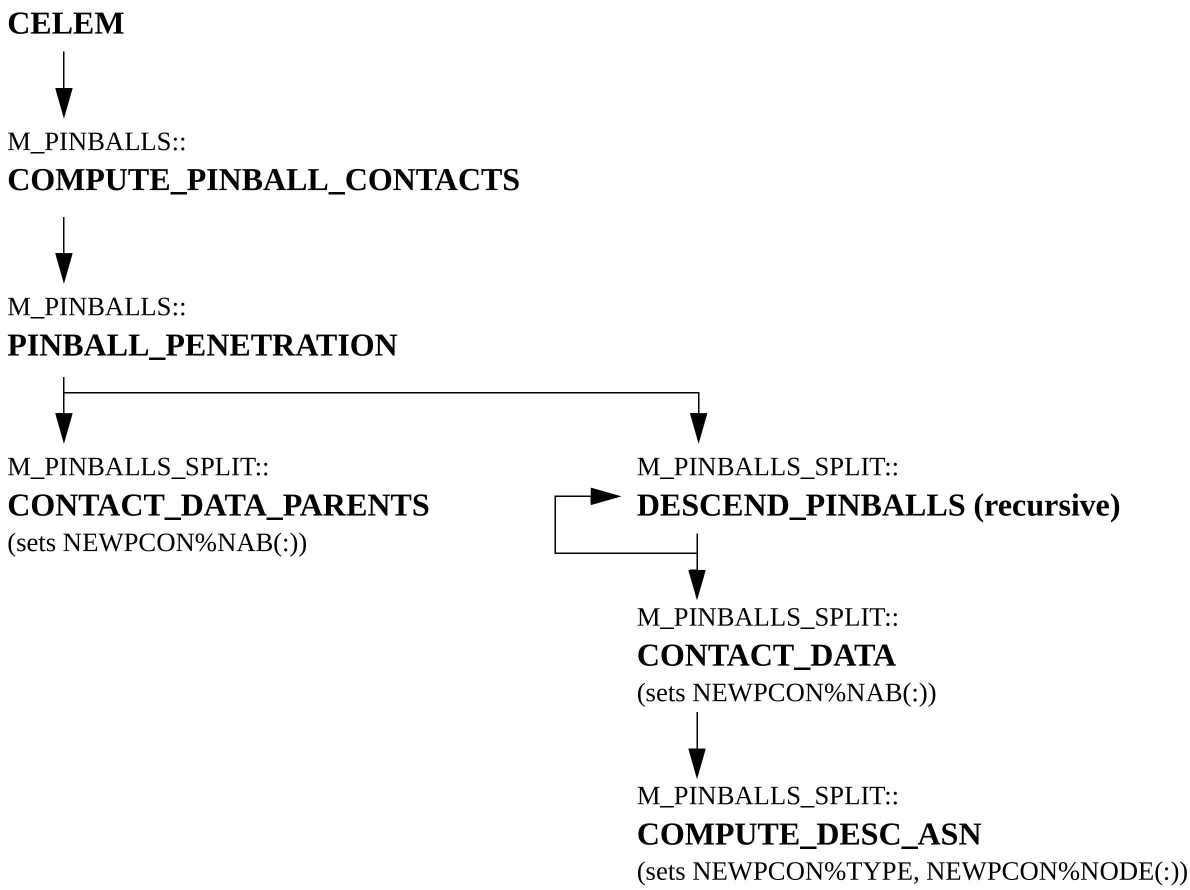


Figure 39 - Flowchart of pinball contact calculations.

4.1 Penetration direction (implemented version)

The algorithm for the penetration direction actually implemented in the code is slightly different from the one listed in Section 2.11.3, because of the particular code structure (see Figure 39) and is reported below.

Two pinballs A, B penetrate each other if, according to (2):

$$\|\underline{C}_{AB}\| < R_A + R_B \,. \tag{89}$$

With reference to the flowchart in Figure 39, this is checked **on the parent pinballs** in the subroutine M_PINBALLS::PINBALL_PENETRATION. When this condition is satisfied, we distinguish between two cases:

- If no hierarchic pinballs are foreseen in the calculation, so that both (parent) pinballs are already at the maximum level of refinement prescribed by the user, then a new pinball contact (PCONTACT) between parent pinballs is created by calling the subroutine M_PINBALLS_SPLIT:: CONTACT_DATA_PARENTS.
- Else, one or both pinballs are further refined (recursively) and penetration is then checked on their descendents, see M_PINBALLS_SPLIT::DESCEND_PINBALLS, until either no penetration is detected any more (so that the search terminates without generating any contact for the current couple of pinballs), or we arrive at the maximum level for both pinballs, and in the latter case a new pinball contact (PCONTACT) is created by calling the subroutine M_PINBALLS_SPLIT::CONTACT_DATA.

4.1.1 Penetration between two parent pinballs

In the first case, both contacting pinballs are parent (E) pinballs. For these pinballs the ASN may or may not be defined. The penetration direction $\hat{\underline{n}}_{AB}$ is computed as follows in the subroutine M_PINBALLS_SPLIT::CONTACT_DATA_PARENTS. In principle one would have, from the algorithm of Section 2.11.3:

- If $\|\underline{n}_A\| = \|\underline{n}_B\| = 0$, i.e. if both pinballs have an undefined associated normal, then the penetration direction is the line joining the two pinball centres (see Figure 29):

$$\hat{\underline{n}}_{AB} = \underline{C}_{AB} / \|\underline{C}_{AB}\| \tag{90}$$

- Else, if ($\|\hat{\underline{n}}_A\| = 1$ and $\|\underline{n}_B\| = 0$), or if ($\|\underline{n}_A\| = 0$ and $\|\hat{\underline{n}}_B\| = 1$), i.e. if only one of the two pinballs has a defined normal, then the penetration direction coincides with either $-\hat{\underline{n}}_A$ or $\hat{\underline{n}}_B$, respectively, see Figure 30. This can be treated either as a special case, or by applying the general equation (4), since the result is the same.

- Else $\|\hat{\underline{n}}_A\| = 1$ and $\|\hat{\underline{n}}_B\| = 1$, i.e. both pinballs have a defined (unit) normal. Then in principle the types of the two pinballs should be considered. However, in this case we know that both pinballs are E pinballs. Then the penetration direction is computed with the general expression (4):

$$\hat{\underline{n}}_{AB} = (\underline{n}_B - \underline{n}_A) / \|\underline{n}_B - \underline{n}_A\| \tag{91}$$

In practice, however, we always compute the penetration direction by the general expression (4) or (91) first. This gives the correct result in the last two cases above, while in the first one the result is undetermined because the denominator of (4) is zero. If the computed denominator is below a small tolerance, we use expression (90) to evaluate the penetration direction. This may occur either in the first case listed above (both ASNs are zero) or also with non-zero ASNs in some limit, degenerated case, where the two contacting pinballs have the same or almost the same ASN. In all these cases using (90) provides a better (probably the best possible) guess for the penetration direction.

4.1.2 Penetration between two pinballs of which at least one is a descendent

In the other case, at least one of the two contacting pinballs is a descendent (F, C or V) pinball (for which a non-zero ASN is guaranteed to exist). Then, following the algorithm of Section 2.11.3 the penetration direction $\hat{\underline{n}}_{AB}$ can be computed in principle as follows in the subroutine M_PINBALLS_SPLIT::CONTACT_DATA.

- If ($\|\hat{\underline{n}}_A\| = 1$ and $\|\underline{n}_B\| = 0$), or if ($\|\underline{n}_A\| = 0$ and $\|\hat{\underline{n}}_B\| = 1$), i.e. if only one of the two pinballs has a defined normal, then the penetration direction coincides with either $-\hat{\underline{n}}_A$ or $\hat{\underline{n}}_B$, respectively, see Figure 30. This can be treated either as a special case, or by applying the general equation (4), since the result is the same.
- Else $\|\hat{\underline{n}}_A\| = 1$ and $\|\hat{\underline{n}}_B\| = 1$, i.e. both pinballs have a defined (unit) normal. Then the types of the two pinballs must be considered (recall that in this case at most one of the pinballs can be an E pinball):
  - If one pinball is an E pinball (so that, at this point of the algorithm, the other cannot be an E pinball), then the normal associated with the other pinball (F, C or V) “has the precedence” over this one and determines the penetration direction. This case occurs when the E pinball is a parent (level $L = 0$) pinball associated either with a continuum or with a structural (shell/beam etc.) element (but not to a material point, since at this point of the algorithm the normal is guaranteed to be non-zero), while the other pinball (F, C or V) is necessarily a descendent pinball, i.e. at a level $L > 0$.
  - Else neither A nor B is an E pinball. Then:

* If both pinballs are face pinballs, then the penetration direction is computed with the general expression (4), see Figure 31.
* Else if one pinball is a face pinball and the other is a corner or vertex pinball, then the normal associated with the face pinball "has the precedence" over the other one and determines the penetration direction, see Figure 32.
* Else if both pinballs are corner pinballs, then the penetration direction is computed with the general expression (4).
* Else if one pinball is a corner pinball and the other is a vertex pinball, then the normal associated with the corner pinball "has the precedence" over the other one and determines the penetration direction.
* Else both pinballs are vertex pinballs. Then, the penetration direction is computed with the general expression (4).

However, by assigning numerical codes 0 to 3 to the pinball types E, V, C, F, respectively (as indicated in the description of the `TYPEI` and `TYPEJ` variables in `TYPE PCONTACT` at the beginning of this Section), the procedure can be simplified and is actually implemented as follows:

- If `TYPEI` is equal to `TYPEJ`, then the general expression (4) is used to compute the penetration direction. Note that at this point of the procedure this can occur only if both pinballs are descendent pinballs, since we cannot have two element (parent) pinballs here.
- Else the two types are different. The type with the maximum numerical index "has the precedence" over the other one and the penetration direction is taken equal to the ASN of the pinball with the higher type.
- In both above cases we check whether the computed penetration vector (before normalization) is physically meaningful. If its length is below a small tolerance, we use expression (90) instead to evaluate the penetration direction.

It is easily verified that this last procedure coincides with the second point of the more complex procedure listed at the beginning of this paragraph (both pinballs have a defined normal), and covers all of its sub-points. But it also works for the first point of the mentioned procedure, i.e. in the case that only one of the two pinballs has a defined normal. In fact, the pinball with undefined normal can only be an E pinball, so that the other pinball (necessarily a descendent pinball at this point of the algorithm) has the precedence in determining the penetration direction.

The situation is summarized in the following Table, where G represents the general expression (4) or (73). In abscissa is represented the type of the first pinball (both as a mnemonic letter and as a numeric code, see the TYPE PCONTACT at the beginning of this Section) and in ordinate the type of the second pinball. Obviously, the table is symmetric with respect to the diagonal.

	0 E	1 V	2 C	3 F
0 E	G	V	C	F
1 V	V	G	C	F
2 C	C	C	G	F
3 F	F	F	F	G

Table 3 - Precedence rules in the contact between two pinballs of which at least one is a descendent.

4.2 Calculation of the ASN for a descendent pinball

We now detail the calculation of the ASN for a descendent pinball, which is performed in subroutine M_PINBALLS_SPLIT::COMPUTE_DESC_ASN (see the flowchart in Figure 39). The routine also fills up the TYPEI/TYPEJ variable and the NODEI(:)/NODEJ(:) array for the PCONTACT, see beginning of this Section. Let DP be the descendent pinball and let IEL be the element to which it is associated (ancestor element of DP). The routine is only called for proper descendents (level $L > 0$), since for parent pinballs the ASN has been already computed (at the beginning of each time step) in another routine (M_PINBALLS::UPDATE_PINBALLS).

We distinguish several cases, according to the “class” of the element IEL to which DP is associated: material point, continuum element, shell element or beam/bar element.

4.2.1 Material point

For a material point no descendent pinballs are admitted, so the routine should not be called. In any way, the ASN to a material point is always zero so it need not be computed.

4.2.2 Continuum element

For a continuum element, we consider the external (to the mesh) faces of the element IEL “touched” (so to say) by DP, i.e. the external faces to which DP is adjacent. Recall that descendent pinballs are built only along the (external) faces of continuum elements and not in the interior of such elements. This information can be readily extracted from the IFACE(:) array (see derived type DESCENDENT_PINBALL at the beginning of this Section), which is filled up elsewhere. Only the strictly positive values, indicating an external face to the mesh, are retained, because (for continuum elements) 0 values indicate an internal sub-face to the element and negative values an internal face to the mesh. The complete set of definitions of the IFACE array is shown in Figure 40, which is taken from reference [13] (Figure 215 on page 269 of that report) and edited to update the terminology: the "corner" (C) and "side" (S) pinballs of reference [13] correspond respectively to vertex (V) and corner (C) pinballs in the present report, while face (F) pinballs are named the same in both reports.

Then:

- If DP touches <u>only one</u> external face of IEL, then DP is a face (F) pinball, and its ASN coincides with the normal to the face concerned. The associated nodes list are simply the nodes of the face.
- If DP touches <u>two</u> external faces of IEL, then we consider two sub-cases.
  - If the space dimension is two, then DP is a vertex (V) pinball, and its ASN coincides with the normal to the node concerned (the common node to the two faces). This normal is contained in the NODAL_ASN(:) array for the node concerned. The associated nodes list contains only this node.
  - Else the space dimension is three, DP is a corner (C) pinball, the node list contains the two nodes common to both faces, and the ASN is the average of the normals to the concerned vertices (NODAL_ASN(:)), projected onto the plane normal to the corner.
- Finally, if DP touches <u>three</u> external faces of IEL (which can occur only in 3D), then DP is a vertex (V) pinball, associated with the common node to the three faces. The nodes list contains only this node and the associated ASN is the corresponding nodal normal in NODAL_ASN(:).

4.2.3 Shell element

If the element is a <u>shell</u>, then we distinguish two cases according to the space dimension.

The IFACE(:) array for a <u>2D shell</u> contains the following information:

```
*(for the parent pinball (not used here):             iface(:) = (2,1))
* for proper descendent pinballs: for vertex i,       iface(:) = (i,i)
*                                 for a face pinball, iface(:) = (1,0)
```

- If IFACE(2) is zero, the descendent is a face (F) pinball having as ASN the same ASN as its ancestor (0-level) pinball and as associated node list the two nodes of the face.
- Else:
  - If the descendent is near an edge of the current shell element with no associated adjacent elements, then it is a vertex (V) pinball with associated zero ASN and with associated node list the node of the concerned edge.

– Otherwise it is a face (F) pinball having as ASN the same ASN as its ancestor (0-level) pinball and as associated node list the two nodes of the face.

The `IFACE(:)` array for a 3D shell contains the following information:

```
*(for the parent pinball (not used here):            iface(:) = (2,1))
* for proper descendent pinballs: for vertex i,       iface(:) = (i,i)
*                                 for a face pinball, iface(:) = (1,0)
*                for edge (corner) of local nodes j-k, iface(:) = (j,k)
```

- If `IFACE(2)` is zero, the descendent is a face (F) pinball having as ASN the same ASN as its ancestor (0-level) pinball, i.e. the normal to the face, and as associated node list the three or four nodes of the face.
- Else:
  - If the descendent is near a node of the current shell element, then it is a vertex (V) pinball with associated zero ASN and as associated node list the node concerned.
  - Else the descendent is near an edge of the current shell element. If the shell element has no adjacents along this edge, then the descendent is a corner (C) pinball with zero ASN and with associated nodes list the two nodes of the edge.
  - Otherwise, the descendent is a face (F) pinball having as ASN the same ASN as its ancestor (0-level) pinball, and as nodes list the three or four nodes of the face.

4.2.4 Beam/bar element

Finally, let us consider the case of a beam/bar element.

The `IFACE(:)` array for a 2D or 3D beam/bar is similar to that for a 2D shell and contains the following information:

```
*(for the parent pinball (not used here):            iface(:) = (2,1))
* for proper descendent pinballs: for vertex i,       iface(:) = (i,i)
*                                 for a face pinball, iface(:) = (1,0)
```

- If `IFACE(2)` is zero, then we distinguish two cases depending upon space dimension:
  - In 2D the descendent is a face (F) pinball having as ASN the same ASN as its ancestor (0-level) pinball and as associated node list the two nodes of the face.
  - In 3D the descendent is a corner (C) pinball having zero ASN and as associated node list the two nodes of the face.
- Else:

– If the descendent is near an edge (node) of the current beam/bar element with no associated adjacent elements, then it is a vertex (V) pinball with associated zero ASN and with associated node list the node of the concerned edge.

– Otherwise, if the space dimension is two (2D), the descendent is a face (F) pinball having as ASN the same ASN as its ancestor (0-level) pinball and as associated node list the two nodes of the face.

– Else the space dimension is three (3D), and the descendent is a corner (C) pinball having zero ASN and as associated node list the two nodes of the face.

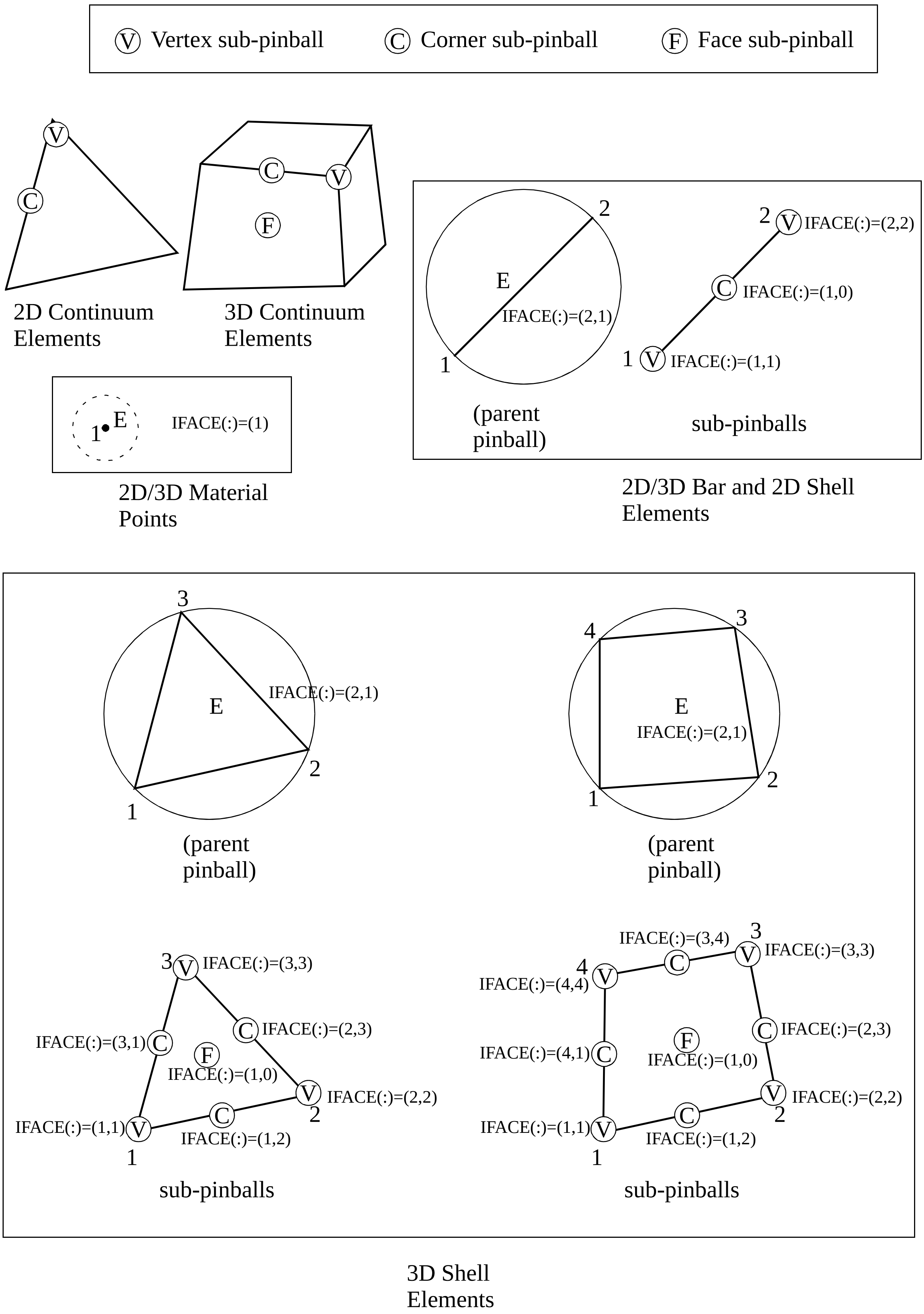


Figure 40 - Definition of the IFACE(:) array for the various pinball types (edited from [13]).

5. Numerical examples

The algorithms described in the previous Section are illustrated by means of numerical examples. All the input files of the examples proposed are listed in the Appendix.

5.1 Visualization of the ASNs

A first set of tests shows the resulting nodal ASNs, the parent pinball ASNs and the descendent pinball ASNS (where appropriate) in various cases. They are listed in Table 4 and described hereafter.

Name	Mesh	Contact parameters	Description
VIDE01	1 CAR1	MLEV 2	Single-element continuum in 2D
VIDE02	1 TRIA	MLEV 2	Single-element continuum in 2D
VIDE05	1 CUBE	MLEV 2	Single-element continuum in 3D
VIDE07	2 CAR1	MLEV 2	Two-element continuum in 2D
ASNO01	3 CAR1 2 TRIA	MLEV 0	Heterogeneous continuum mesh in 2D
ASNO02	4 CUB8	MLEV 2	Bar impact in 3D
ASNO03	400 CAR1	MLEV 0	Impacting bodies in 2D
ASNO04	360 CAR1	MLEV 0	Hollow body in 2D

Table 4 - Tests to show the nodal, parent pinball and descendent pinball ASNs.

VIDE01

This test shows the nodal ASNs and the (parent) pinball ASNs for a stand-alone 2D quadrilateral (CAR1). A hierarchic pinball of level 2 (`MLEV 2`) is embedded in the element and the option `OPTI PINS VIDE` is activated in order to consider all descendents in contact and to be able to visualize the final descendents and their ASNs.

The left part of Figure 41 shows the parent pinball (`PARE`) and the nodal ASNs (`NASN`). The (parent) ASN (`PASN`) is zero as expected in this case because this is a stand-alone element. The right part of the Figure shows the final (level 2) descendents and the associated ASNs (`DASN`).

VIDE02

This test shows the nodal ASNs and the (parent) pinball ASNs for a stand-alone 2D triangle (TRIA). A hierarchic pinball of level 2 (`MLEV 2`) is embedded in the element and the option `OPTI PINS VIDE` is activated in order to consider all descendents in contact and to be able to visualize the final descendents and their ASNs.

The left part of Figure 41 shows the parent pinball and the nodal ASNs. The (parent) ASN is zero as expected in this case because this is a stand-alone element. The right part of the Figure shows the final (level 2) descendents and the associated ASNs.

VIDE05

This test shows the nodal ASNs and the (parent) pinball ASNs for a stand-alone 3D hexahedron (CUBE). A hierarchic pinball of level 2 (`MLEV 2`) is embedded in the element and the option `OPTI PINS VIDE` is activated in order to consider all descendents in contact and to be able to visualize the final descendents and their ASNs.

The left part of Figure 43 shows the parent pinball and the nodal ASNs. The (parent) ASN is zero as expected in this case because this is a stand-alone element. The right part of the Figure shows the final (level 2) descendents and the associated ASNs.

VIDE07

This test shows the nodal ASNs and the (parent) pinball ASNs for a mesh of two 2D quadrilaterals (CAR1). Hierarchic pinballs of level 2 (`MLEV 2`) are embedded in the elements and the option `OPTI PINS VIDE` is activated in order to consider all descendents in contact and to be able to visualize the final descendents and their ASNs.

The left part of Figure 41 shows the parent pinballs, the nodal ASNs and the (parent) pinball ASNs. The (parent) ASNs are no longer zero, as expected in this case. The right part of the Figure shows the final (level 2) descendents and the associated ASNs.

ASNO01

This test considers the mesh of Figure 1, composed by three quadrilaterals and two triangles in 2D. Parent pinballs are embedded in all the elements.

The left part of Figure 45 shows the parent pinballs, the nodal ASNs and the (parent) pinball ASNs on the whole mesh. The right part of the Figure shows the same quantities but on a subset of the mesh (to check visualization implementation).

ASNO02

This test considers a simplified bar impact in 3D. The mesh is composed by two CUB8 hexahedral element for each bar. Hierarchic pinballs of level 2 (`MLEV 2`) are embedded only in the elements which are likely to come into contact (one element for each bar).

The upper part of Figure 46 shows the parent pinballs, the nodal ASNs, the (parent) pinball ASNs, the contacting descendents and their ASNs. The bottom part of the Figure shows details of the descendent ASNs and of the resulting contact normal directions. Note that the contact normals (in green) are all vertical, i.e. in the direction of the impact, although the same is not true for the descendent ASNs.

ASNO03

This test considers the impact between two rectangular bodies in 2D. The mesh is composed by two hundred CAR1 quadrilateral elements for each rectangular body. Parent pinballs of level 0 are embedded only in the elements which are likely to come into contact, i.e. along the surface of the two bodies.

The left part of Figure 47 shows the parent pinballs, the right part shows the (parent) pinball ASNs and the nodal ASNs.

ASNO04

This test considers a hollow circular body in 2D. The mesh is composed by 360 CAR1 quadrilateral elements. Parent pinballs of level 0 are embedded only in the elements along the surface (both internal and external) of the body.

The left part of Figure 48 shows the parent pinballs, the right part shows the (parent) pinball ASNs and the nodal ASNs.

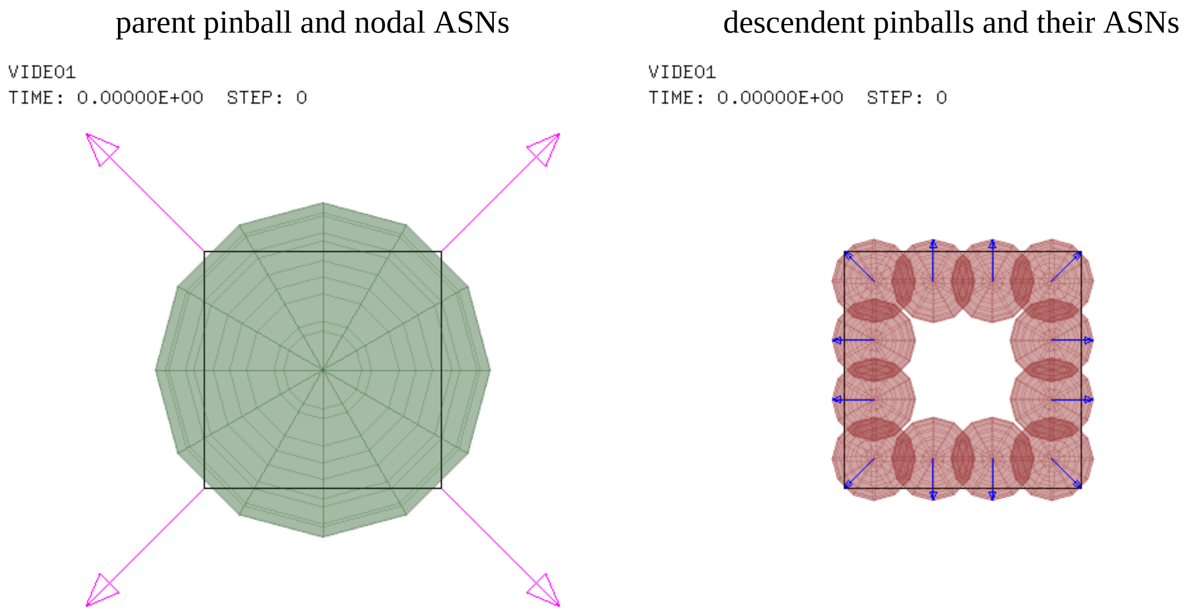


Figure 41 - ASNs in case VIDE01.

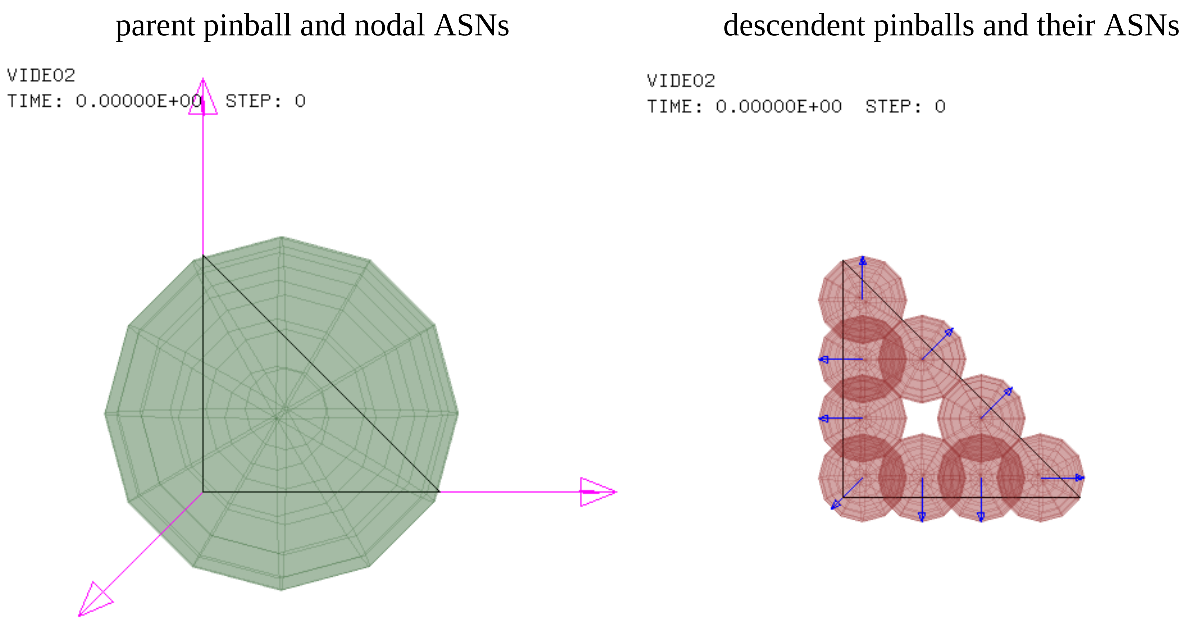


Figure 42 - ASNs in case VIDE02.

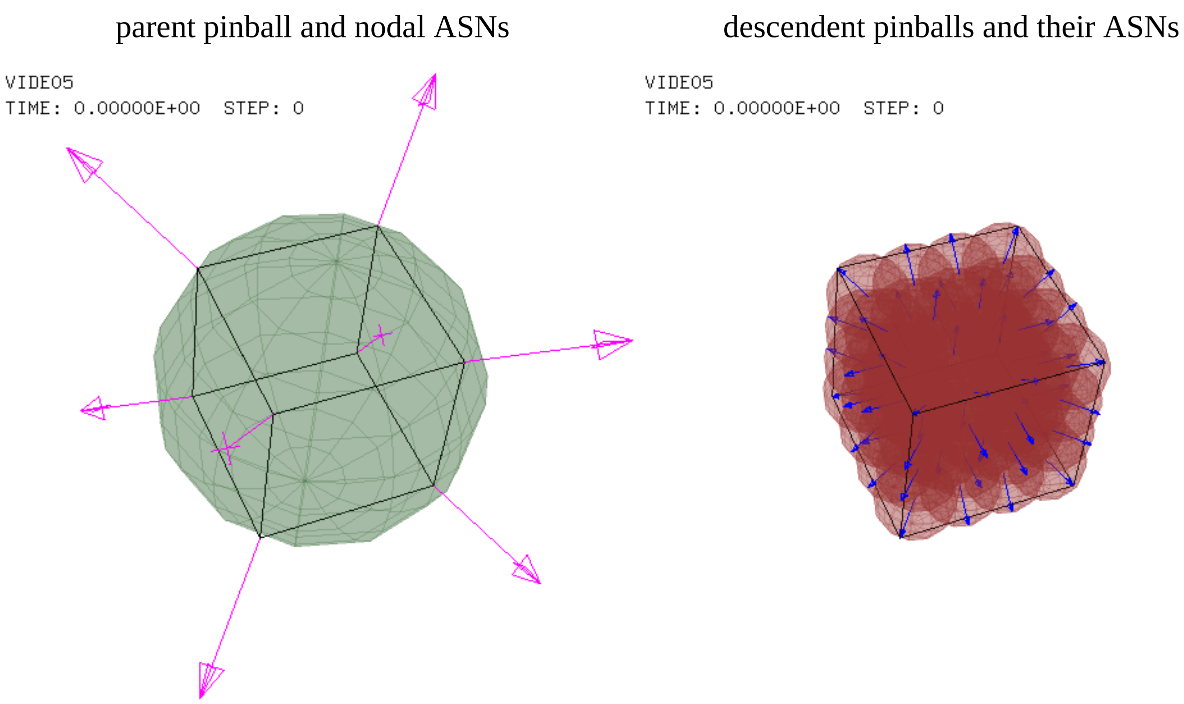


Figure 43 - ASNs in case VIDE05.

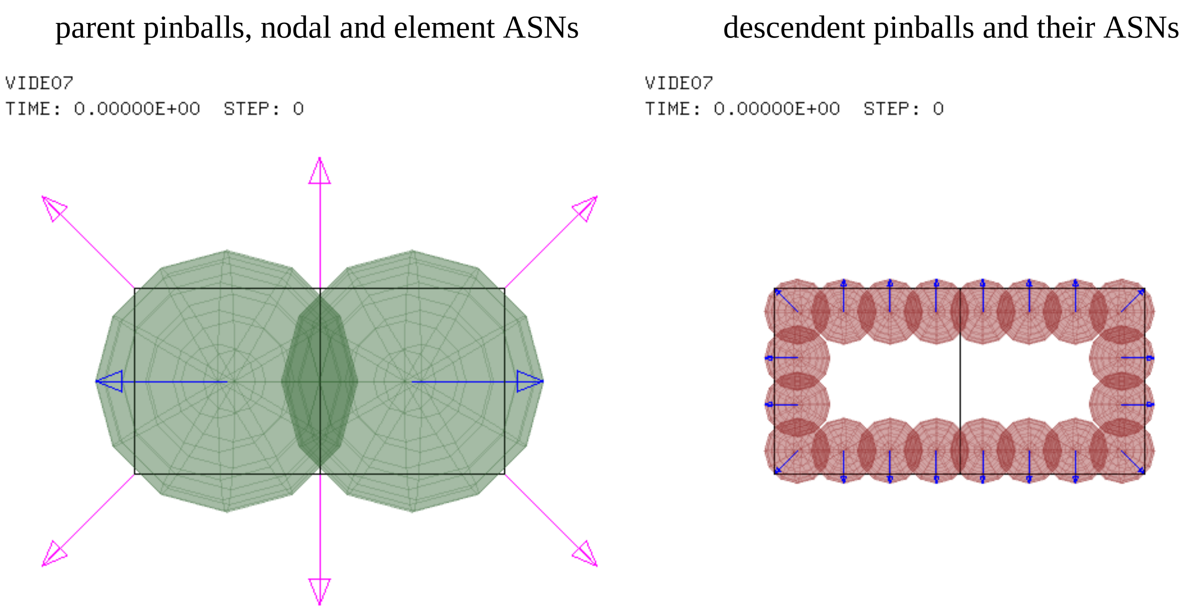


Figure 44 - ASNs in case VIDE07.

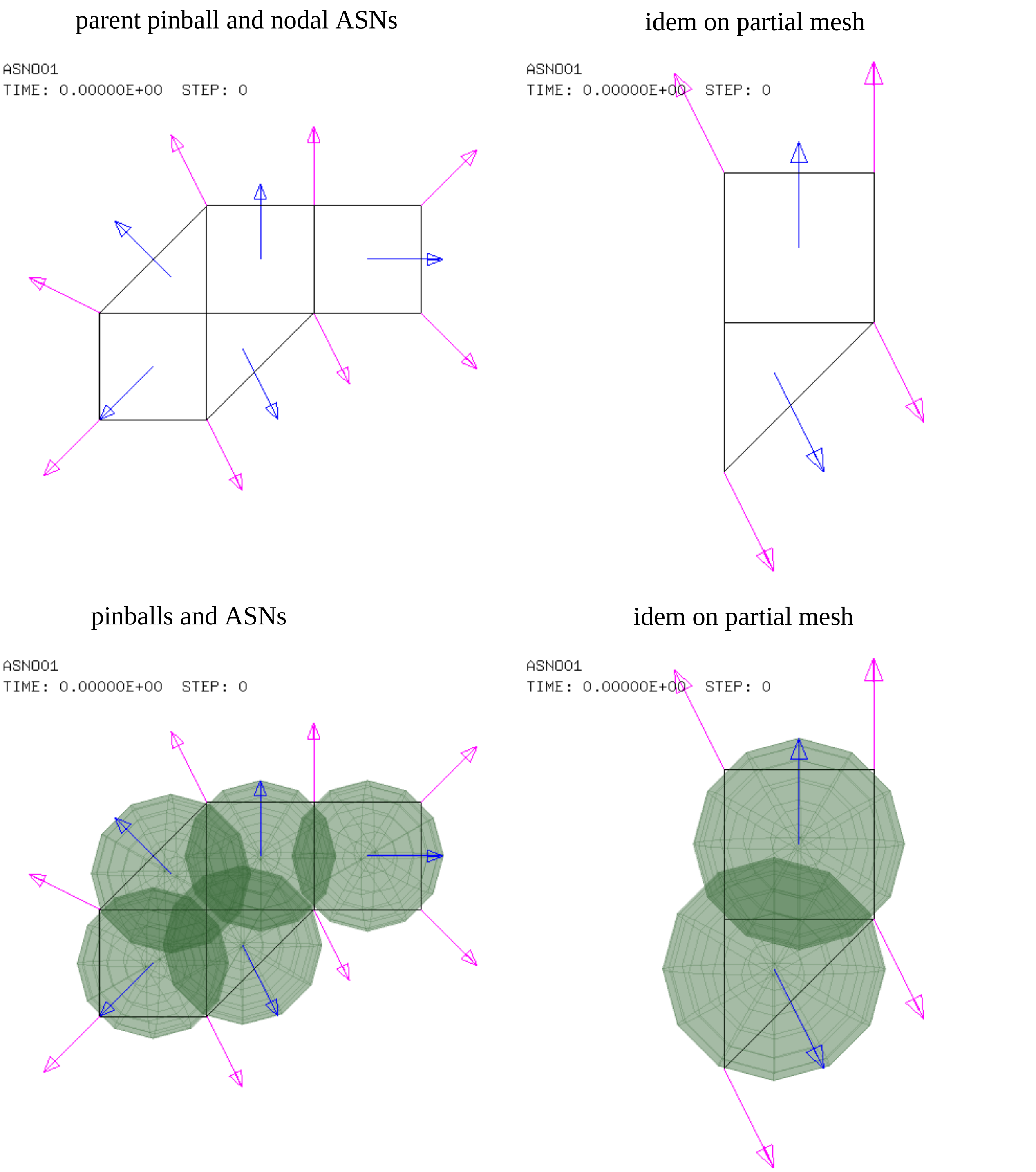


Figure 45 - ASNs in case ASNO01.

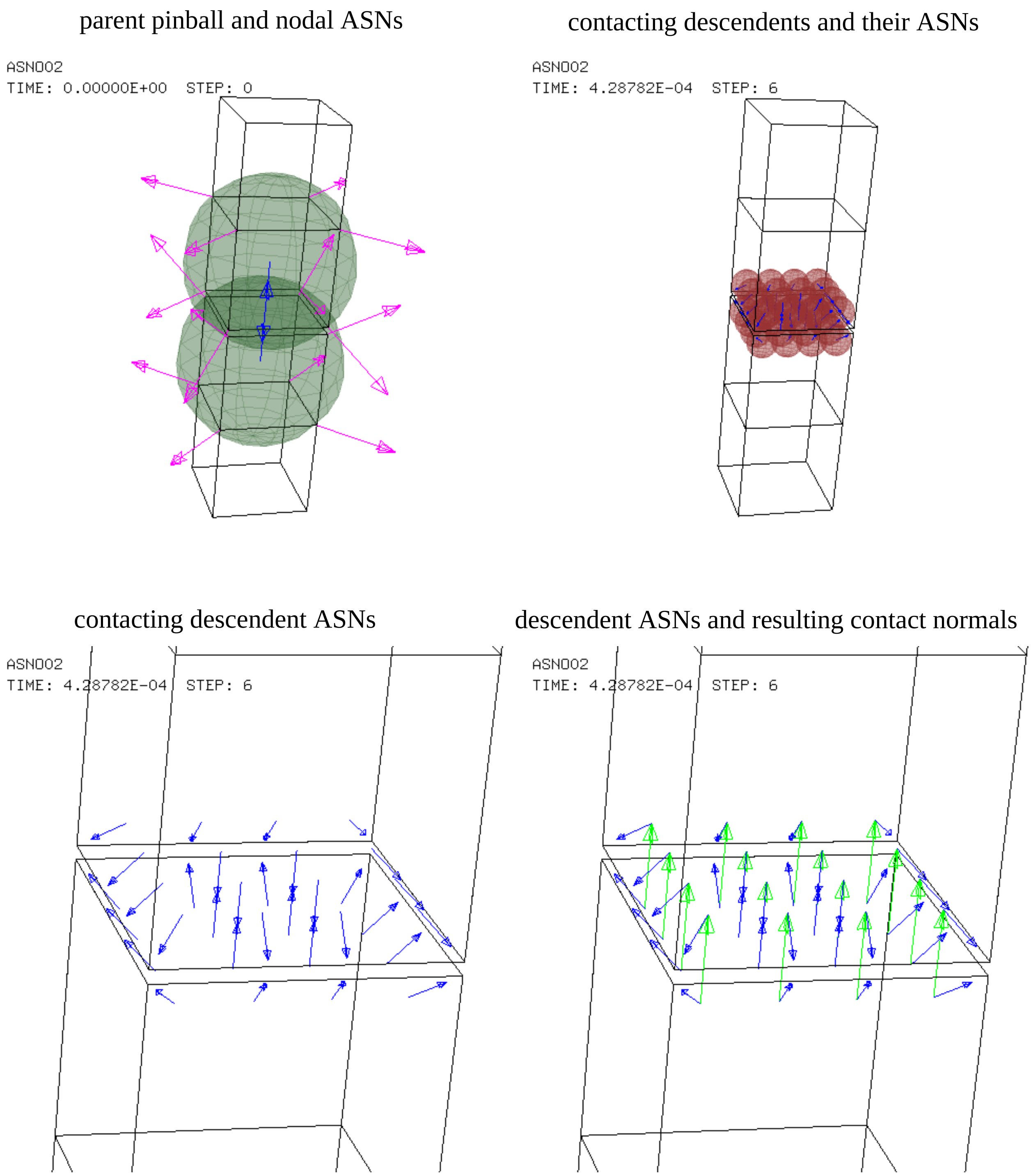


Figure 46 - ASNs in case ASNO02.

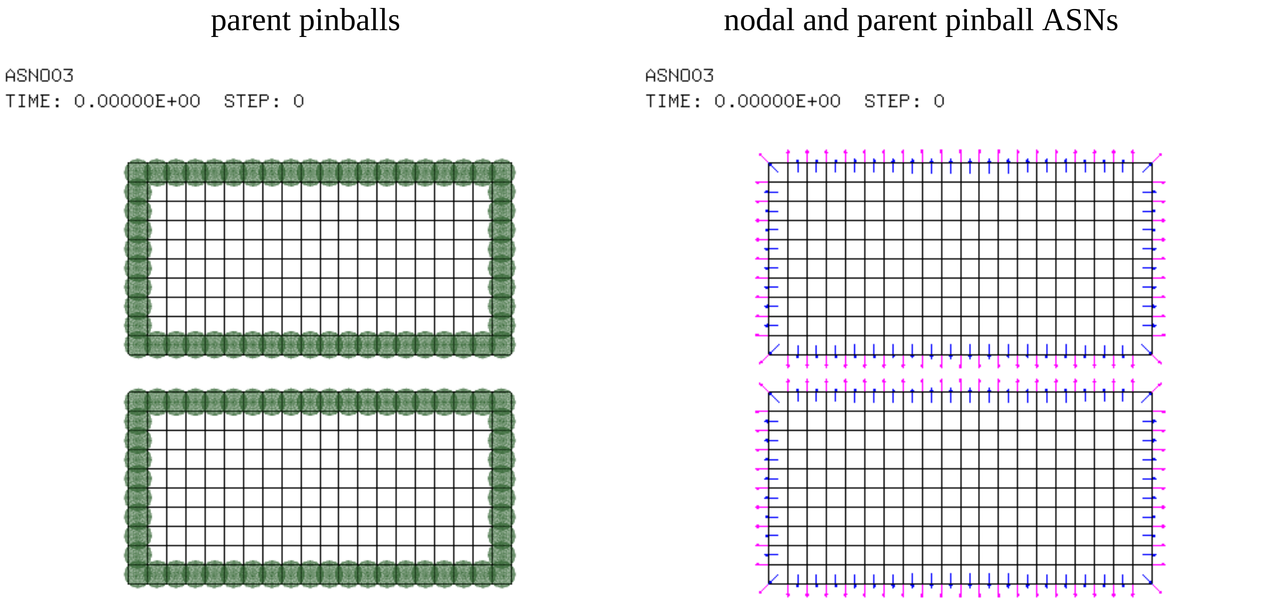


Figure 47 - ASNs in case ASNO03.

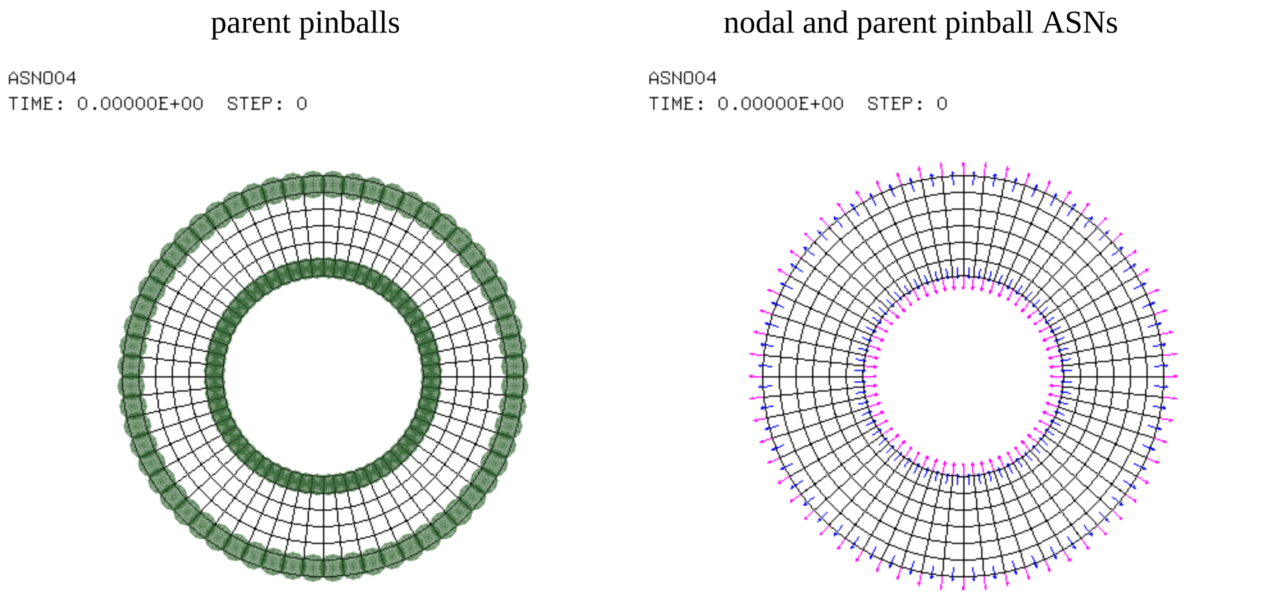


Figure 48 - ASNs in case ASNO04.

5.2 Visualization of the ASNs and of the (sub-)pinball types

The next set of tests checks the ASNs and the resulting (sub-)pinball types in simple cases for the different element types. They are listed in Table 5 and described hereafter.

Name	Mesh	Contact parameters	Description
ASNC01	4 CAR1	2 BODY MLEV 0	Contact with continuum elements in 2D
ASNC02	4 CAR1	2 BODY MLEV 1	Contact with continuum elements in 2D
ASNC03	4 CAR1	2 BODY MLEV 2	Contact with continuum elements in 2D
ASNC04	8 CAR1	2 BODY MLEV 0	Contact with continuum elements in 2D
ASNC05	5 BARR	1 BODY MLEV 0	Contact with bar elements in 2D
ASNC06	5 BARR	1 BODY MLEV 1	Contact with bar elements in 2D
ASNC07	5 Q4GS	1 BODY MLEV 0	Contact with shell elements in 3D
ASNC08	5 Q4GS	1 BODY MLEV 1	Contact with shell elements in 3D
ASNC09	5 Q4GS	1 BODY MLEV 2	Contact with shell elements in 3D
ASNC10	5 Q4GS	1 BODY MLEV 3	Contact with shell elements in 3D

Table 5 - Tests to show the ASNs and the (sub-)pinball types.

ASNC01

Two blocks each formed by two CAR1 quadrilateral elements are in contact. Zero level (`MLEV 0`, by default) pinballs are specified for each block. Figure 49 shows the contacting pinballs, which in this case coincide with the parent pinballs (left), the nodal ASNs, parent ASNs and contact normals (right) and the descendent ASNs (which in this case coincide with the parent ASNs). Nodal ASNs have the expected directions. The types of contacting pinballs (`TYPEI`, `TYPEJ`) are 0 and the nodes list (`NODEI(:)`, `NODEJ(:)`) are also 0, as expected since these quantities are defined only for sub-pinballs (at level $L > 0$).

ASNC02

This test is similar to ASNC01 but here we take `MLEV 1` pinballs. Figure 50 shows the contacting pinballs (left), the nodal ASNs, parent ASNs and contact normals (right) and the descendent ASNs. Nodal ASNs have the expected directions. The types of contacting pinballs and the nodes list are as expected: magenta indicates V pinballs and blue indicates F pinballs.

ASNC03

This test is similar to ASNC01 but here we take `MLEV 2` pinballs. Figure 51 shows the contacting pinballs (left), the nodal ASNs, parent ASNs and contact normals (right) and the descendent ASNs.

Nodal ASNs have the expected directions. The types of contacting pinballs and the nodes list are as expected: magenta indicates V pinballs and blue indicates F pinballs.

ASNC04

This test is similar to ASNC01 but we have two layers of CAR1 elements in each contacting body, instead of just one, for a total of 8 CAR1 elements. This is to avoid "degenerated" parent normals (the horizontal blue arrows in Figure 49 for the ASNC01 case). Figure 52 shows the contacting pinballs (left), the nodal ASNs, parent ASNs and contact normals (right) and the descendent ASNs. Nodal ASNs have the expected directions. The types of contacting pinballs and the nodes list are as expected: magenta indicates V pinballs and blue indicates F pinballs. The parent normals now have a more physical direction.

ASNC05

This test checks ASNs in 2D bar elements (BARR). Zero-level pinballs (`MLEV 0` by default) are specified. Figure 53 shows the parent pinballs, the parent ASNs and the nodal ASNs (which are 0 in this case) in the left part, the contacting descendents (of which there are none in this case, since there is only one `BODY`) in the right part.

ASNC06

This test is identical to ASNC05 but uses `MLEV 1`. The `OPTI PINS VIDE` option is added in order to activate contacts (with the 0-pinball) in all descendent pinballs, so that they can be visualized. Figure 54 shows the parent pinballs, the parent ASNs and the nodal ASNs (which are 0 in this case) in the left part, the contacting descendents and their ASNs in the right part.

As expected, contacting descendents are either of vertex (V) type, represented in magenta and with a 0 associated ASN in this case, or of face (F) type, represented in blue.

ASNC07

This test is the 3D version of case ASNC05, and uses Q4GS quadrilateral shell elements. Zero-level pinballs (`MLEV 0` by default) are specified. Figure 55 shows the parent pinballs, the parent ASNs and the nodal ASNs (which are 0 in this case) in the left part, the contacting descendents (of which there are none in this case, since there is only one `BODY`) in the right part.

ASNC08

This test is identical to ASNC08 but uses `MLEV 1`. The `OPTI PINS VIDE` option is added in order to activate contacts (with the 0-pinball) in all descendent pinballs, so that they can be visualized. Figure 56 shows the parent pinballs, the parent ASNs and the nodal ASNs (which are 0 in this case) in the left part, the contacting descendents and their ASNs in the right part. As expected, contacting descendents are all of vertex (V) type, represented in magenta and with a 0 associated ASN in this case.

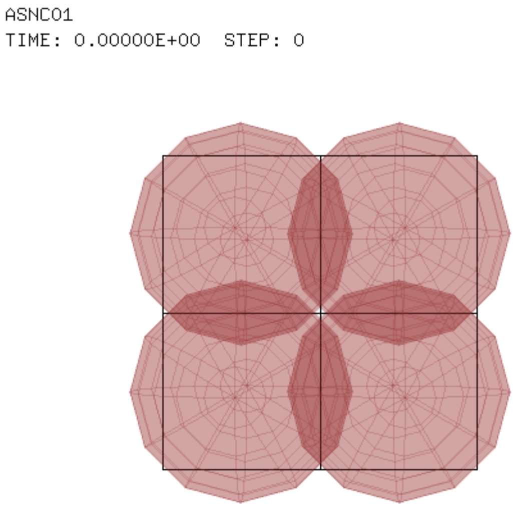
ASNC09

This test is identical to ASNC08 but uses `MLEV 2`. The `OPTI PINS VIDE` option is added in order to activate contacts (with the 0-pinball) in all descendent pinballs, so that they can be visualized. Figure 57 shows the parent pinballs, the parent ASNs and the nodal ASNs (which are 0 in this case) in the left part, the contacting descendents and their ASNs in the right part. As expected, contacting descendents are either of vertex (V) type, represented in magenta and with a 0 associated ASN, or of corner (C) type, represented in cyan and with a 0 associated ASN, or of face (F) type, represented in blue and with an associated non-zero ASN.

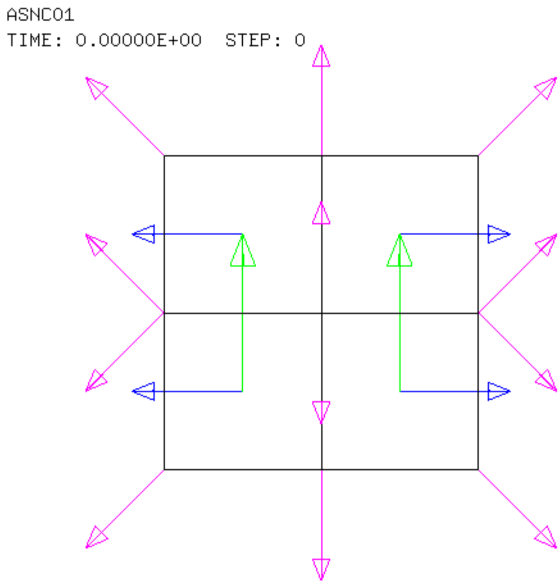
ASNC10

This test is identical to ASNC08 but uses `MLEV 3`. The `OPTI PINS VIDE` option is added in order to activate contacts (with the 0-pinball) in all descendent pinballs, so that they can be visualized. Figure 58 shows the parent pinballs, the parent ASNs and the nodal ASNs. As expected, contacting descendents are either of vertex (V) type, represented in magenta and with a 0 associated ASN, or of corner (C) type, represented in cyan and with a 0 associated ASN, or of face (F) type, represented in blue and with an associated non-zero ASN.

contacting pinballs (parent pinballs in this case)



nodal ASNs, parent ASNs and contact normals



descendent ASNs (parent ASNs in this case)

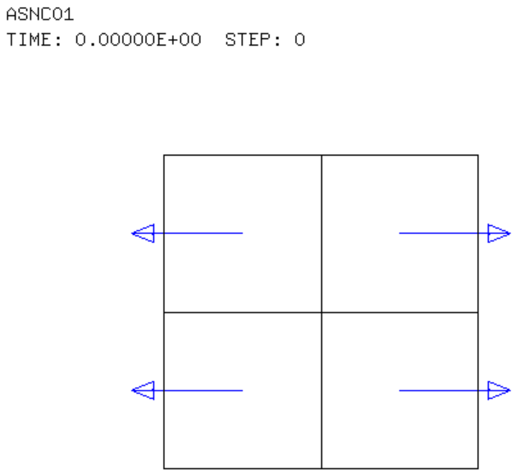
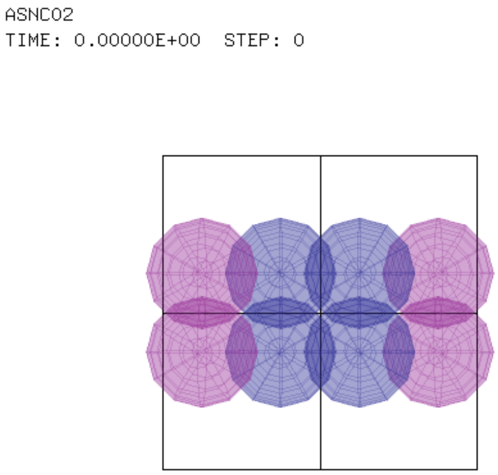
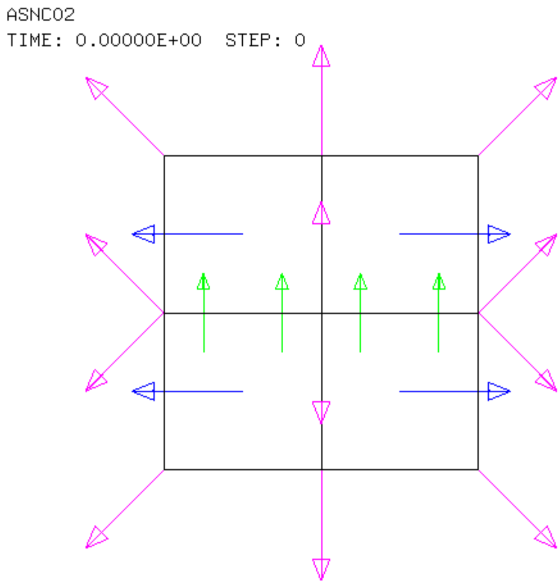


Figure 49 - ASNs in case ASNC01.

contacting descendent pinballs



nodal ASNs, parent ASNs and contact normals



descendent ASNs

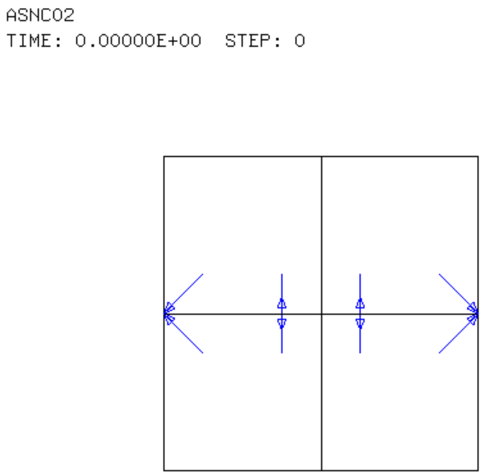


Figure 50 - ASNs in case ASNC02.

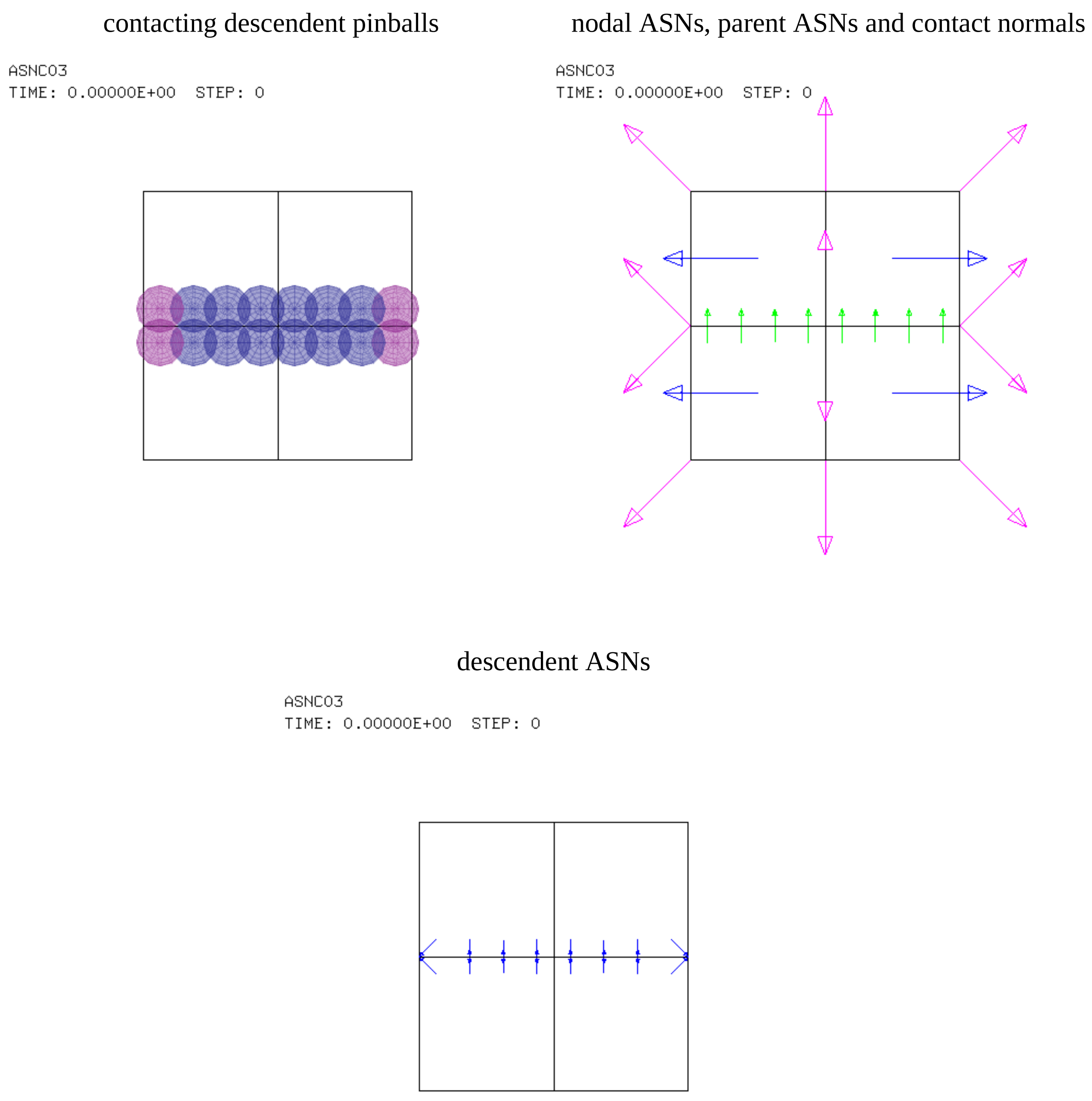
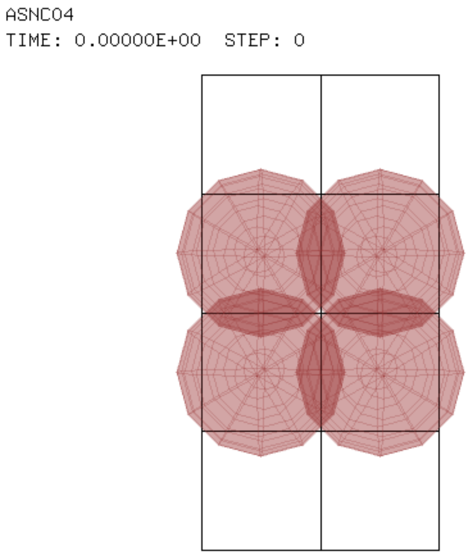
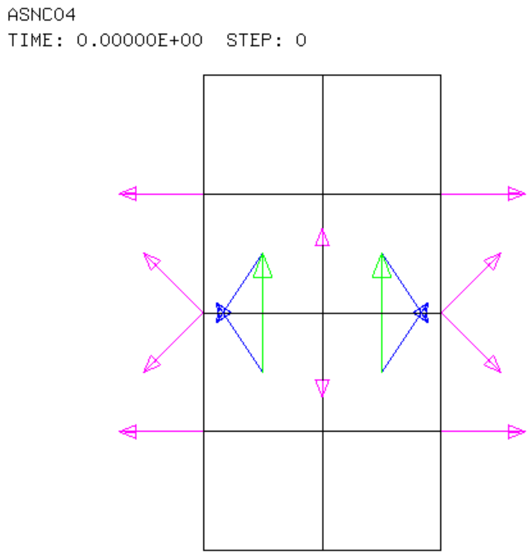


Figure 51 - ASNs in case ASNC03.

contacting pinballs (parent pinballs in this case)



nodal ASNs, parent ASNs and contact normals



descendent ASNs (parent ASNs in this case)

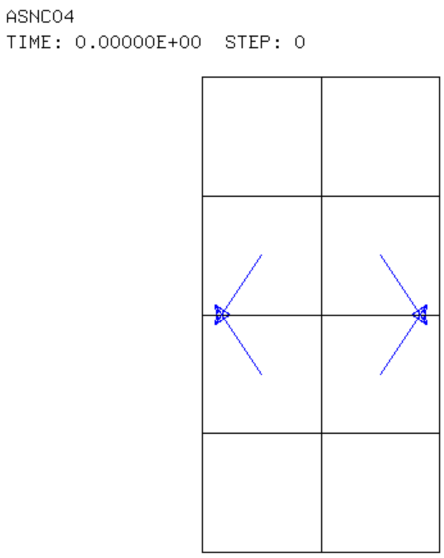


Figure 52 - ASNs in case ASNC04.

parent pinballs, parent ASNs and nodal ASNs (0)   contacting descendents (none) and desc. ASNs

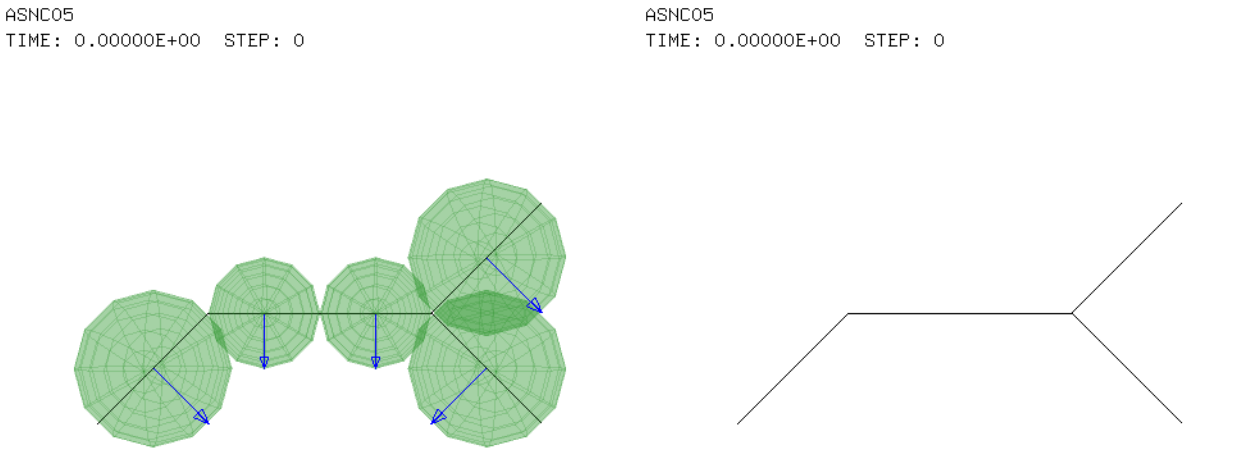


Figure 53 - ASNs in case ASNC05.

parent pinballs, parent ASNs and nodal ASNs (0)   contacting descendents and descendent ASNs

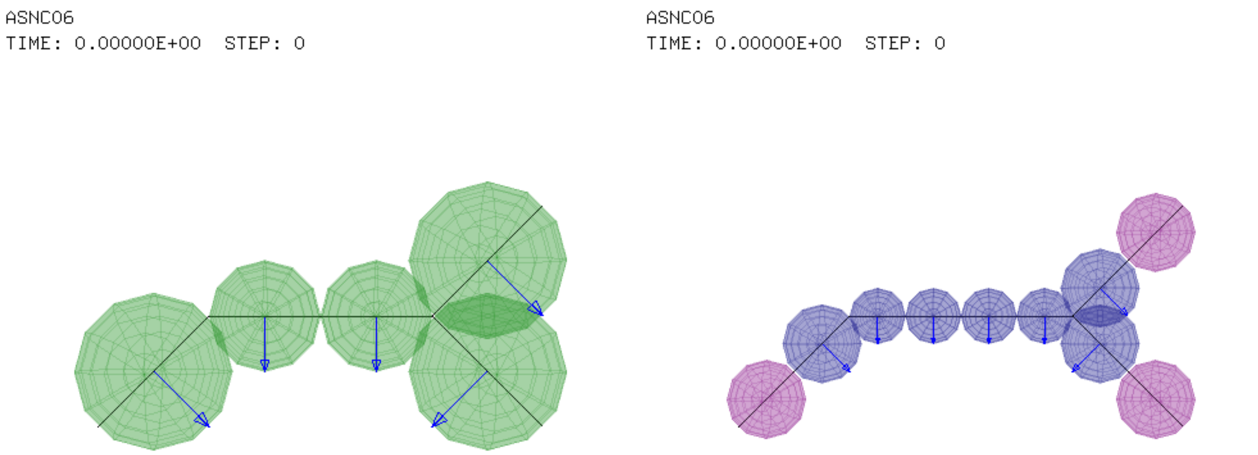


Figure 54 - ASNs in case ASNC06.

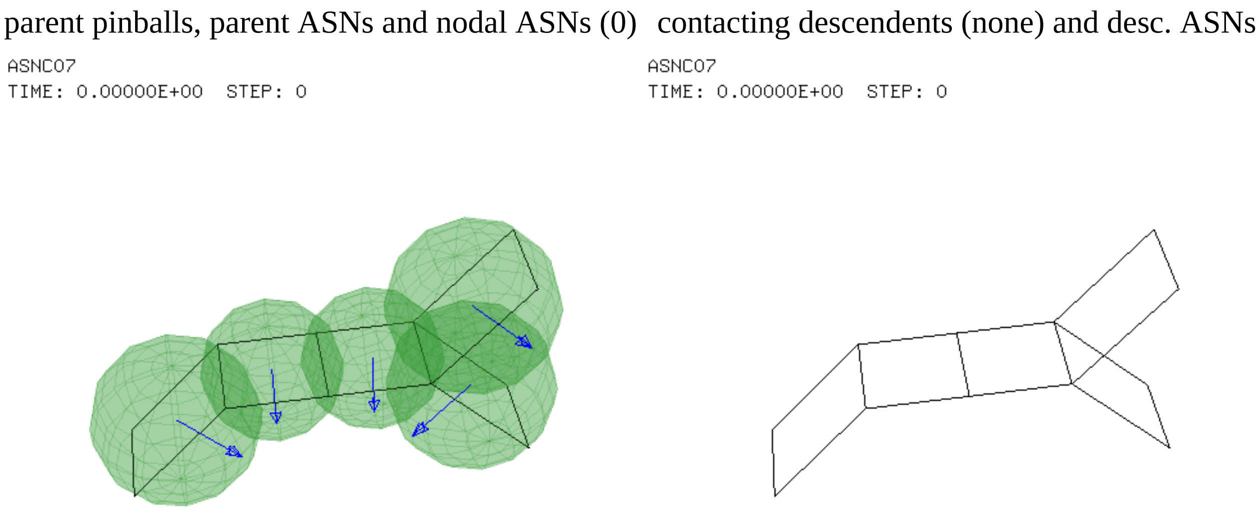


Figure 55 - ASNs in case ASNC07.

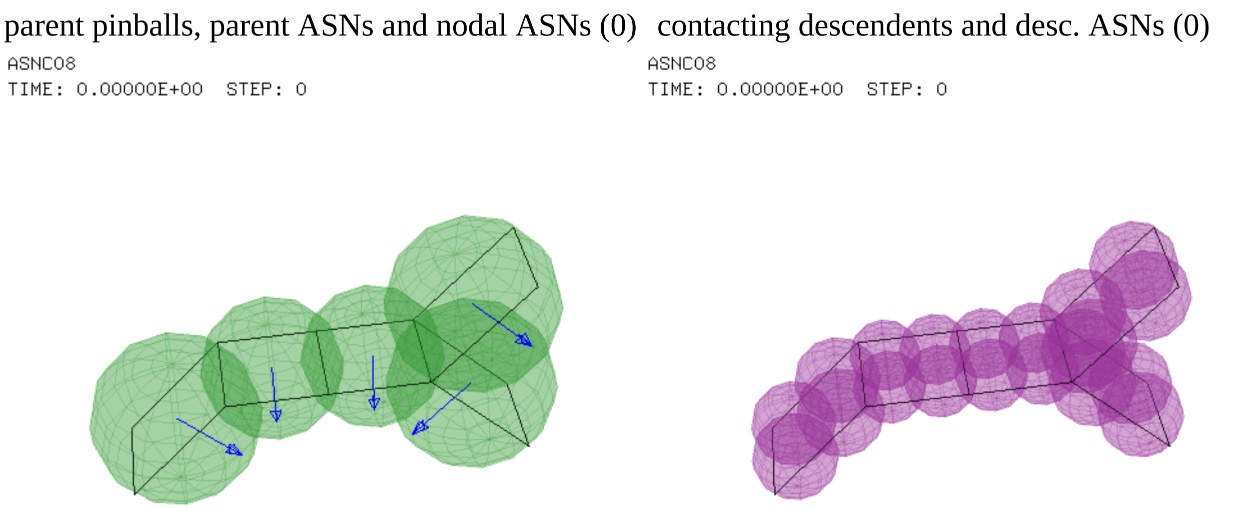


Figure 56 - ASNs in case ASNC08.

parent pinballs, parent ASNs and nodal ASNs (0) contacting descendents and desc. ASNs

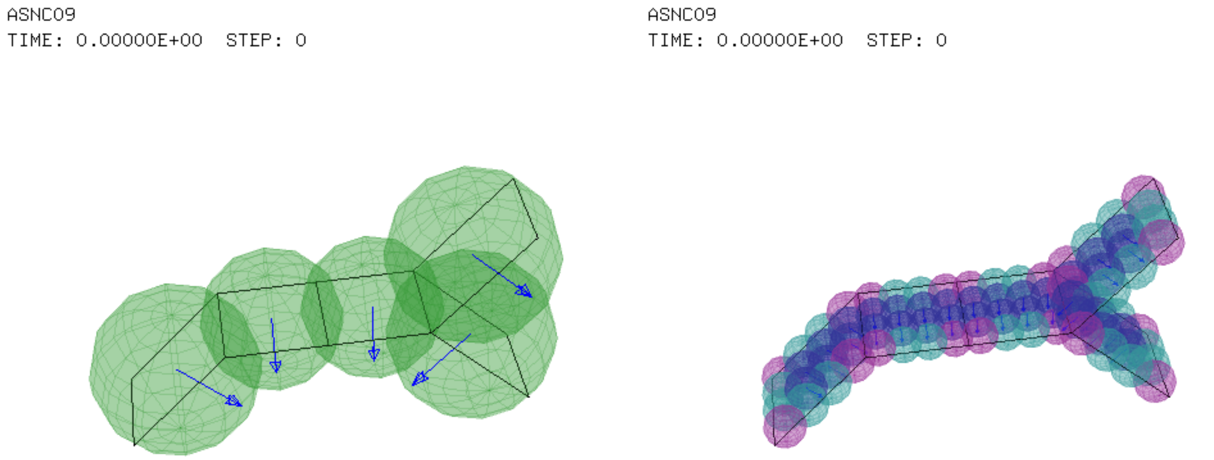


Figure 57 - ASNs in case ASNC09.

parent pinballs, parent ASNs and nodal ASNs (0) contacting descendents and desc. ASNs

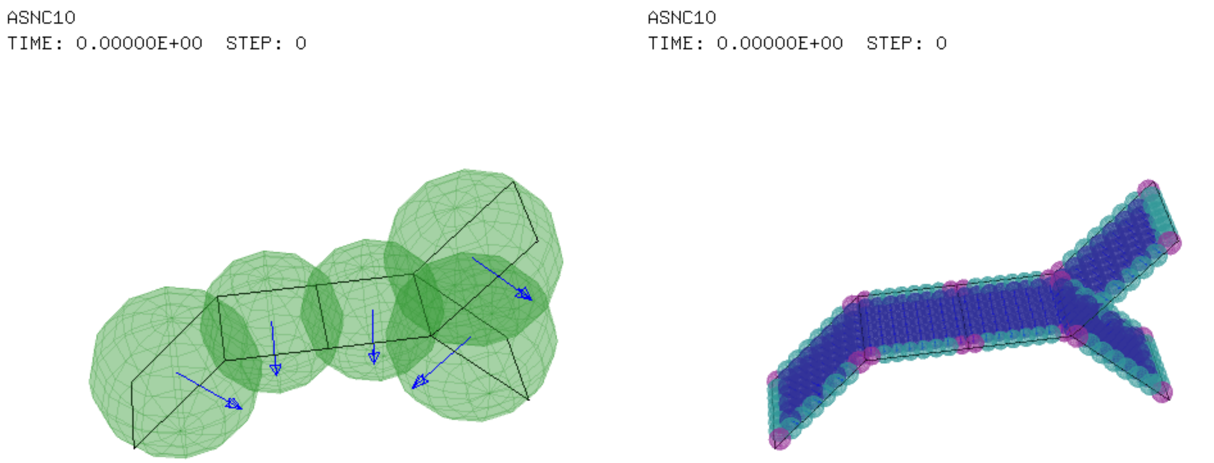


Figure 58 - ASNs in case ASNC10.

5.3 Simple impact tests

The next set of tests shows simple impact simulations.

5.3.1 Impact between continuum elements

We start by considering impacts between continuum elements. They are listed in Table 6 and are described hereafter.

Name	Mesh	Contact parameters	Description
PENE01	2 CAR1	PENA MLEV 0 ASN	single-element impact in 2D
PENE02	4 CAR1	PENA MLEV 0 ASN	two plus two element impact in 2D
PENE03	4 CUBE	PENA MLEV 0 ASN	two plus two element impact in 3D
PENE04	4 CUBE	PENA MLEV 0	two plus two element impact in 3D
PENE05	4 CUBE	PENA MLEV 2 VIDE ASN	two plus two element impact in 3D
PENE06	4 CUBE	PENA MLEV 2 ASN	two plus two element impact in 3D

Table 6 - Simple impact tests between continuum elements.

PENE01

This test considers the impact between two quadrilaterals in 2D, represented by just one CAR1 element each. A parent pinball (level 0) is embedded in each of the elements. A zero initial gap exists between the two elements, leading to a large initial interpenetration of the parent pinballs.

The left part of Figure 59 shows the parent pinballs and the nodal ASNs. The right part of the Figure shows the nodal ASNs, the (parent) ASNs (which are zero in this case, because these are two stand-alone elements) and the contact normal (in green).

Note that in this case the general formula (25) for the contact normal (or for the penetration direction) would give an undefined direction because the two pinballs normals are zero. In such a case the code takes the centers-joining line (see eq. 72) as the best possible guess for the contact direction, as shown in the right part of Figure 59. This is just to avoid a fatal error message. However, it remains that the solution of this contact problem with such a coarse mesh is non-physical, especially if the contact forces are uniformly distributed over all the element's nodes like in the present case. A finer mesh would be needed to obtain plausible results.

Note that in this impact problem (and in the following ones) we systematically specify the `OPTI PINS NORB` keyword in order to completely disable any special treatment of rebound. According to Section 2.9, such treatments are necessary when using the LM method, but not with the penalty method.

PENE02

This test is similar to PENE01 but uses twice longer impacting bars, each discretized by two CAR1 elements. The level of pinballs is 0 and they are embedded only in the two elements that come into contact. An initial gap of 0.42 m is assumed so that the pinballs are not in contact initially. Opposite initial velocities of 50 m/s are assumed.

The left part of Figure 60 shows the parent pinballs, which enter into contact at step 2. The right part of the Figure shows the nodal ASNs and the (parent) pinball ASNs.

Figure 61 shows the nodal displacements of two nodes on the opposite contacting surfaces and Figure 62 shows the corresponding contact forces.

PENE03

This test is the 3D version of PENE01 and uses two CUBE elements for each bar. The level of pinballs is 0 and they are embedded only in the two elements that come into contact. An initial gap of 0.75 m is assumed so that the pinballs are not in contact initially. Opposite initial velocities of 50 m/s are assumed.

The left part of Figure 63 shows the parent pinballs, which enter into contact at step 5. The right part of the Figure shows the nodal ASNs and the (parent) pinball ASNs.

Figure 64 shows the nodal displacements of two nodes on the opposite contacting surfaces and Figure 65 shows the corresponding contact forces.

PENE04

This test is similar to PENE03 but without the `OPTI PINB ASN` option.

The left part of Figure 66 shows the parent pinballs, which enter into contact at step 5. The right part of the Figure shows the nodal ASNs and the (parent) pinball ASNs, which of course are all zero in this case because the ASN model is not activated.

Figure 67 shows the nodal displacements of two nodes on the opposite contacting surfaces and Figure 68 shows the corresponding contact forces.

The results are identical to those of case PENE03 which used the ASN method. This is because the two methods give the same (perfectly vertical) contact normal in this particular case (perfectly aligned impact).

PENE05

This test is similar to PENE03 but with `MLEV 2` and `VIDE` option to show all the descendent pinballs and the associated ASNs.

The left upper part of Figure 69 shows the parent pinballs, the nodal ASNs and the (parent) pinball ASNs. The right upper part of the Figure shows the descendent pinballs and the associated ASNs. The lower part of the Figure shows a detail of the descendent pinball ASNs.

PENE06

This test is similar to PENE05 but without the `VIDE` option, so that a real calculation is performed. The initial gap is 0.2 m.

Figure 70 shows the nodal displacements of two nodes on the opposite contacting surfaces and Figure 71 shows the corresponding contact forces.

parent pinball and nodal ASNs

descendent pinballs and their ASNs

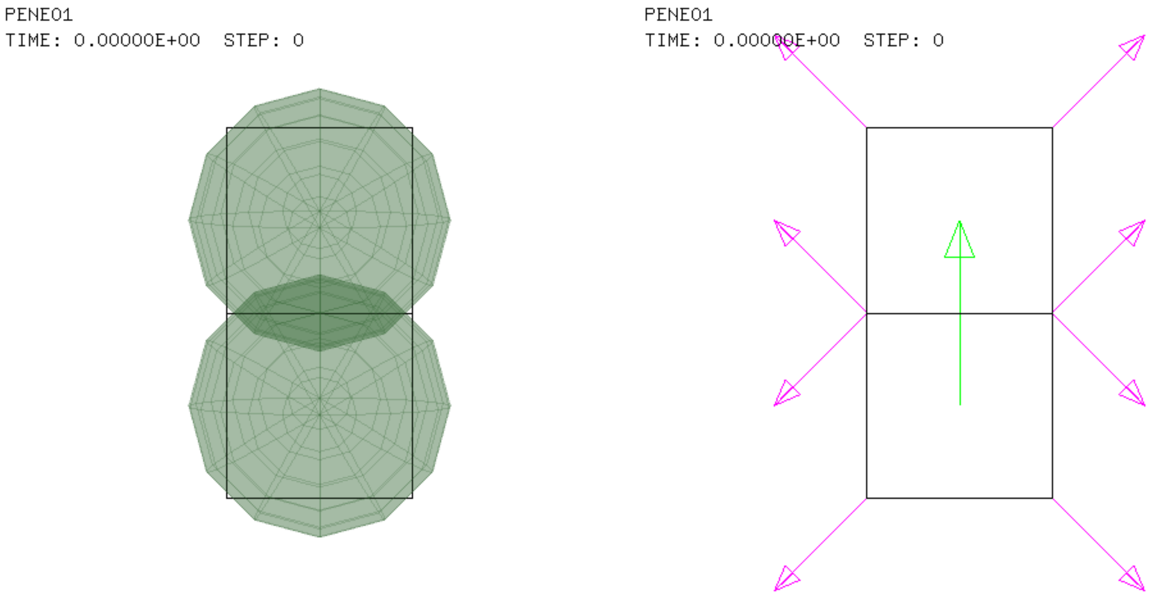


Figure 59 - ASNs in case PENE01.

parent pinballs

nodal and pinball ASNs

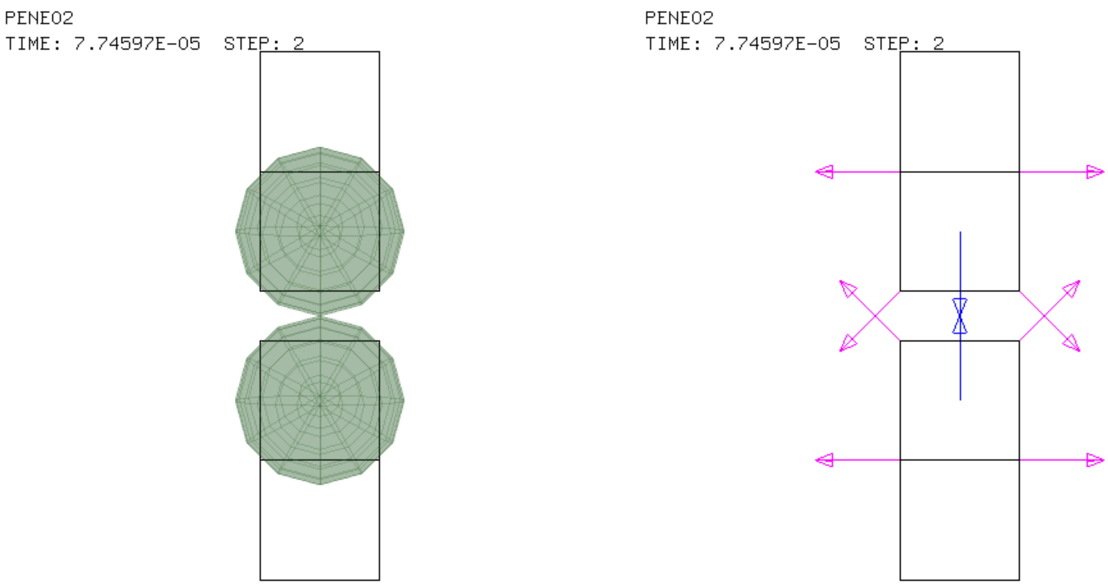


Figure 60 - ASNs in case PENE02.

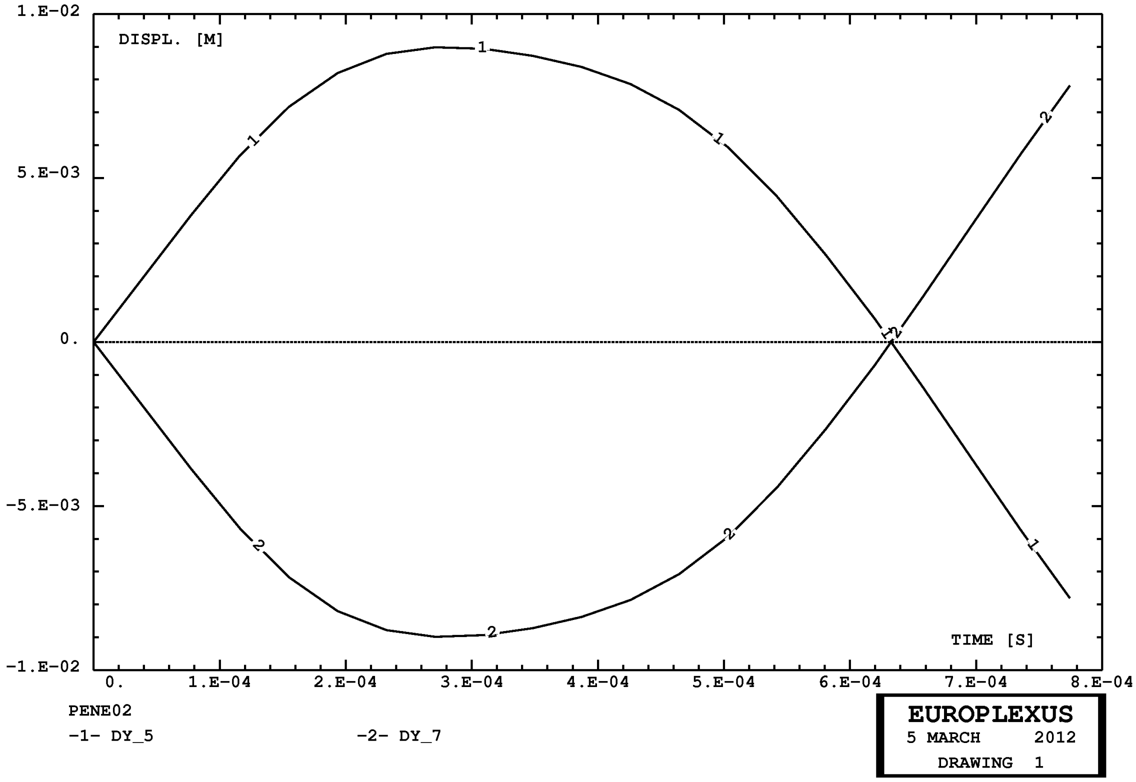


Figure 61 - Displacements in case PENE02.

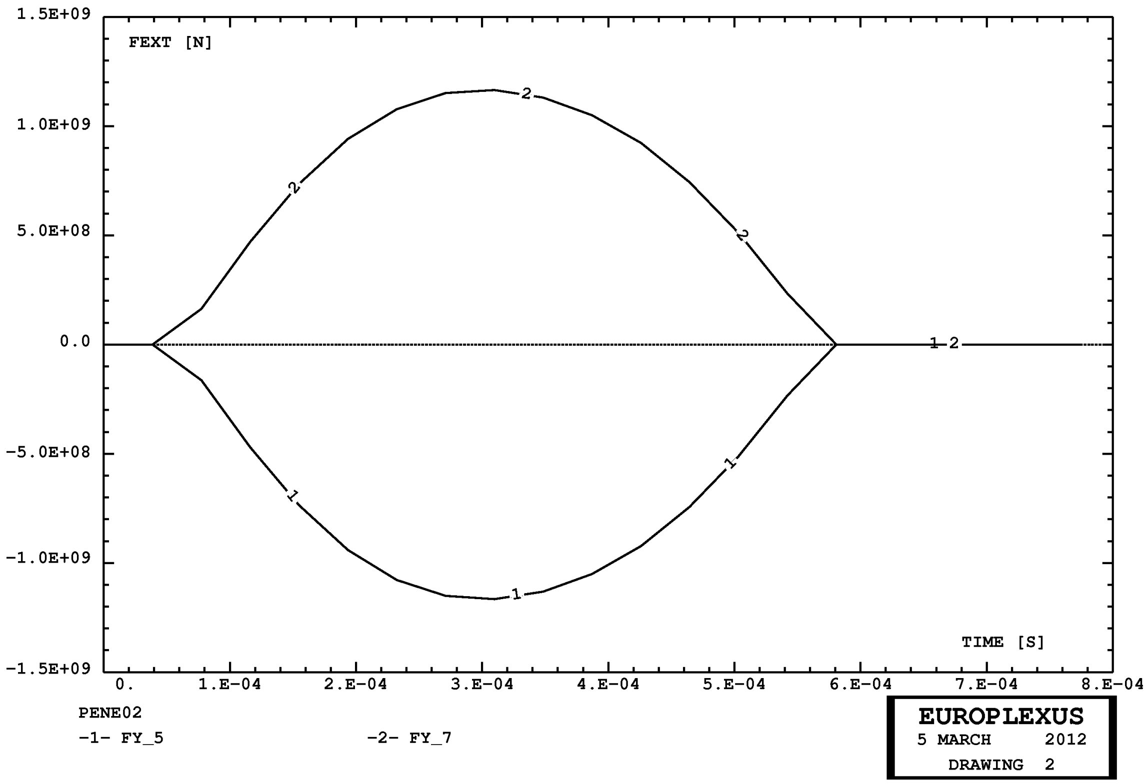


Figure 62 - Contact forces in case PENE02.

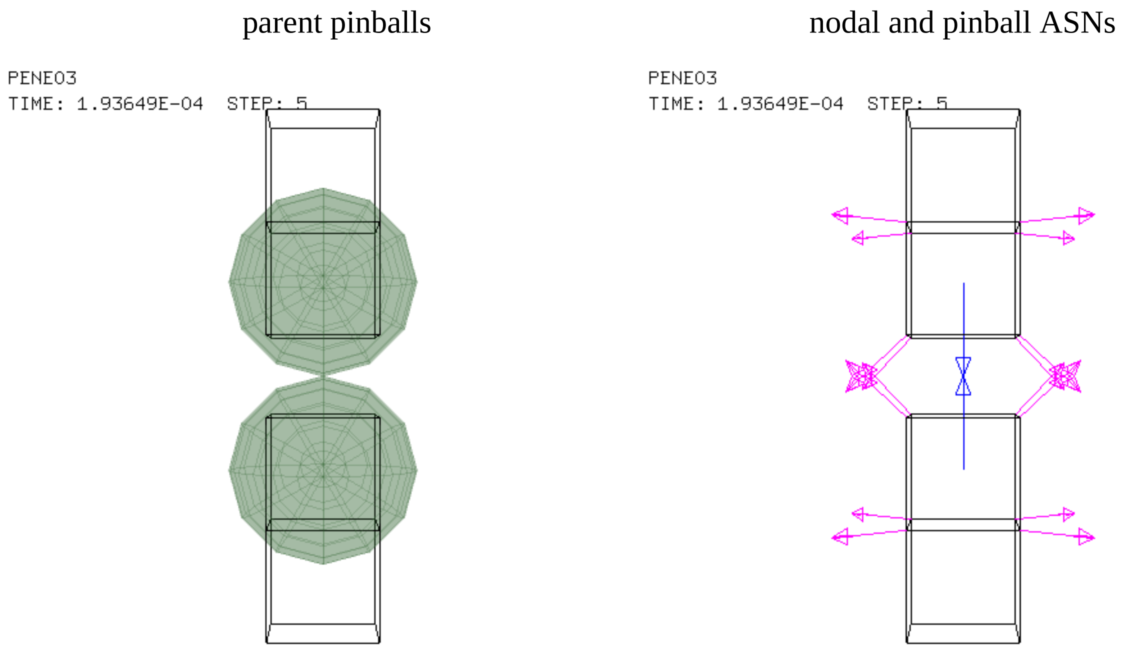


Figure 63 - ASNs in case PENE03.

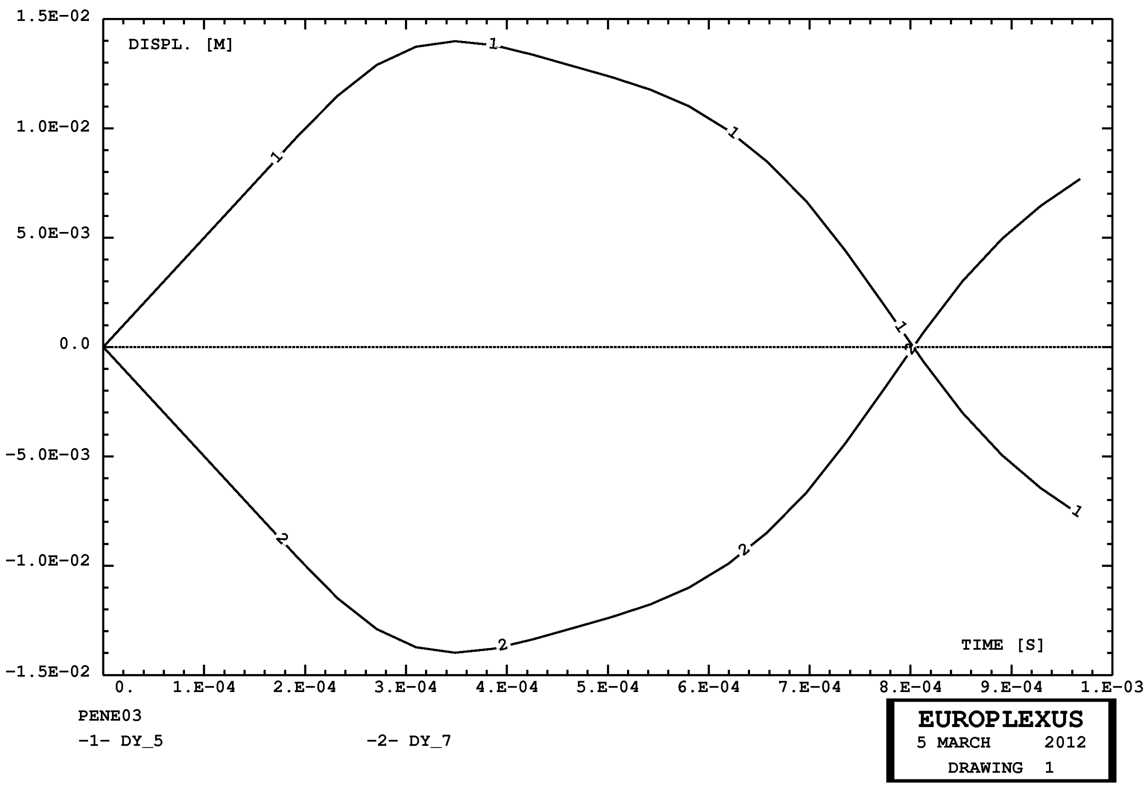


Figure 64 - Displacements in case PENE03.

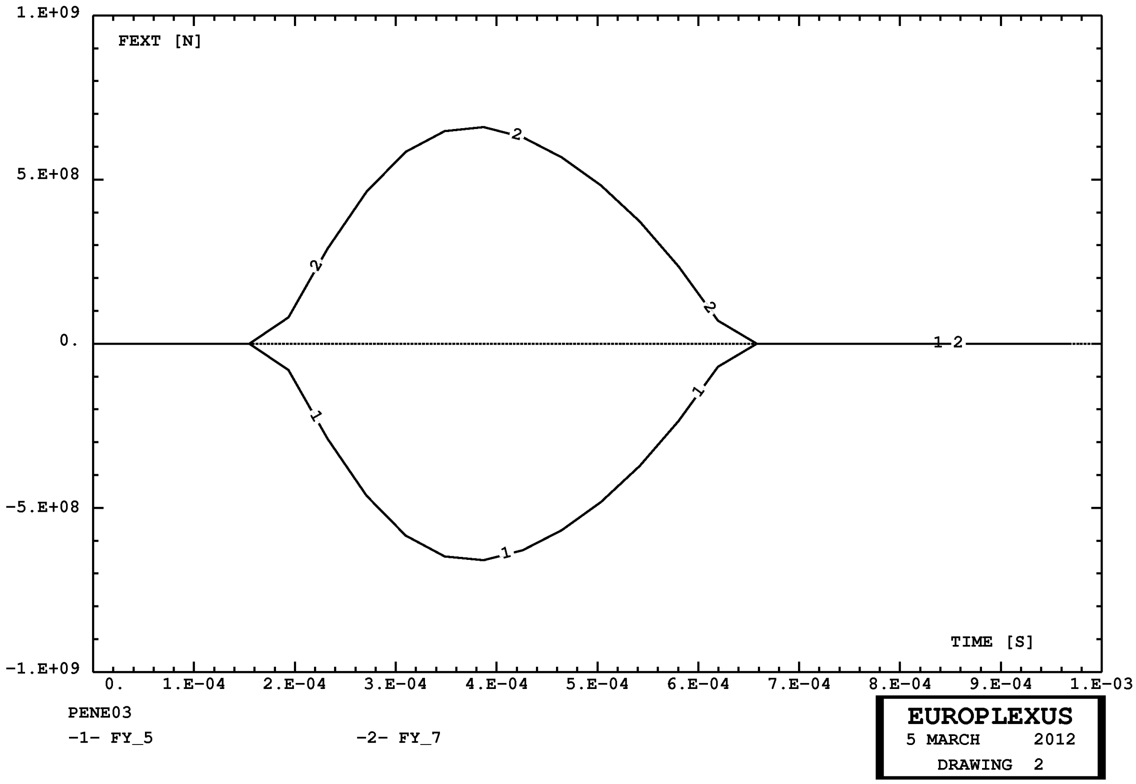


Figure 65 - Contact forces in case PENE03.

parent pinballs

nodal and pinball ASNs

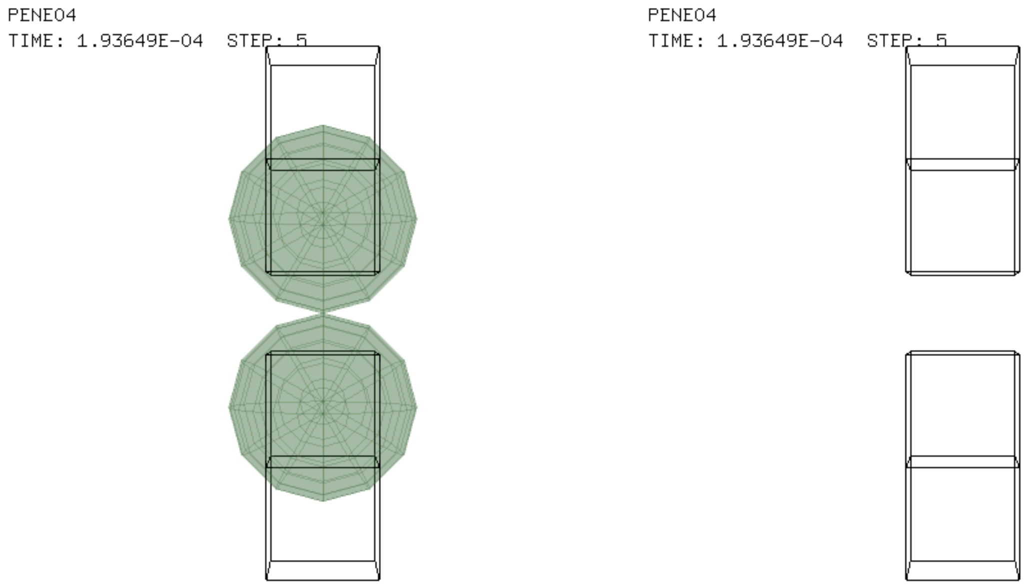


Figure 66 - ASNs in case PENE04.

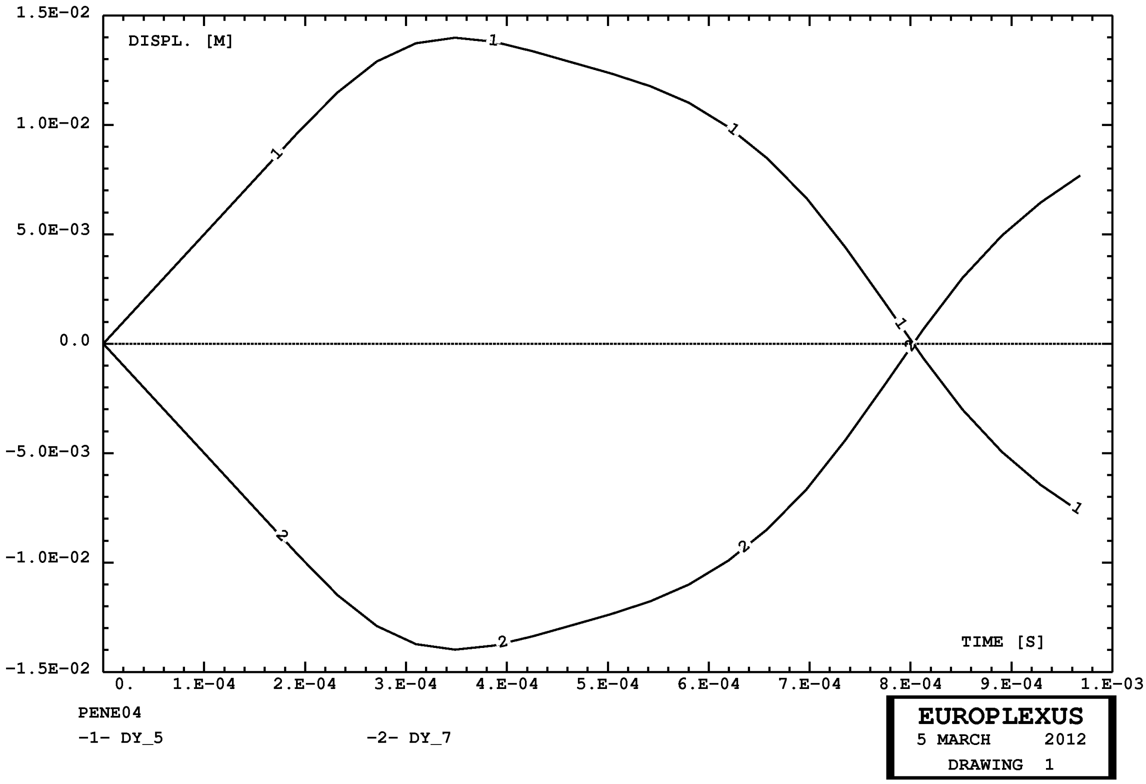


Figure 67 - Displacements in case PENE04.

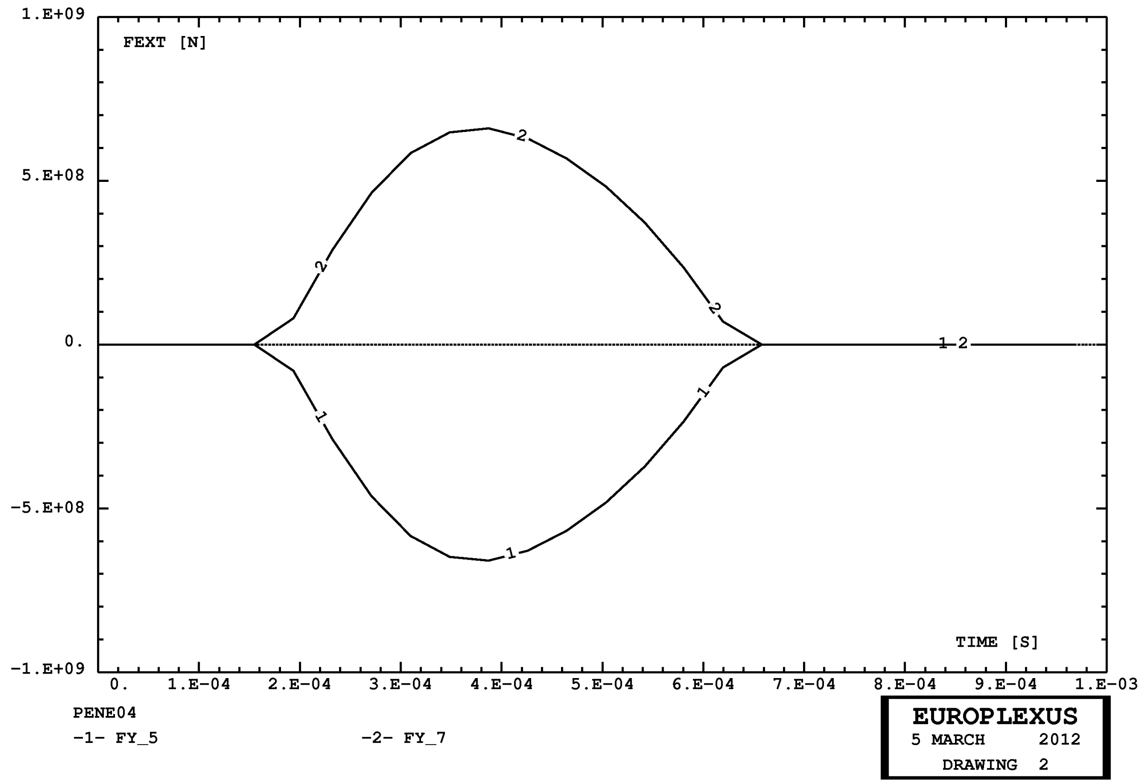
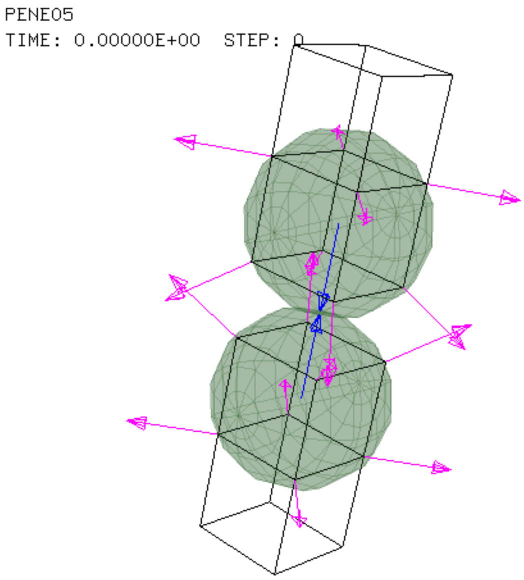
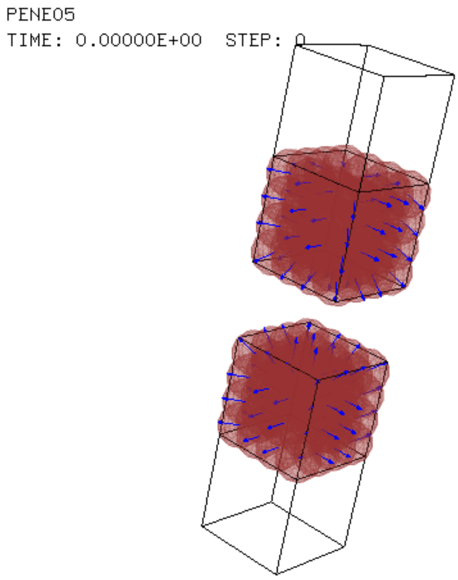


Figure 68 - Contact forces in case PENE04.

parent pinball, nodal and pinball ASNs



descendent pinballs and their ASNs



detail of descendent pinball ASNs

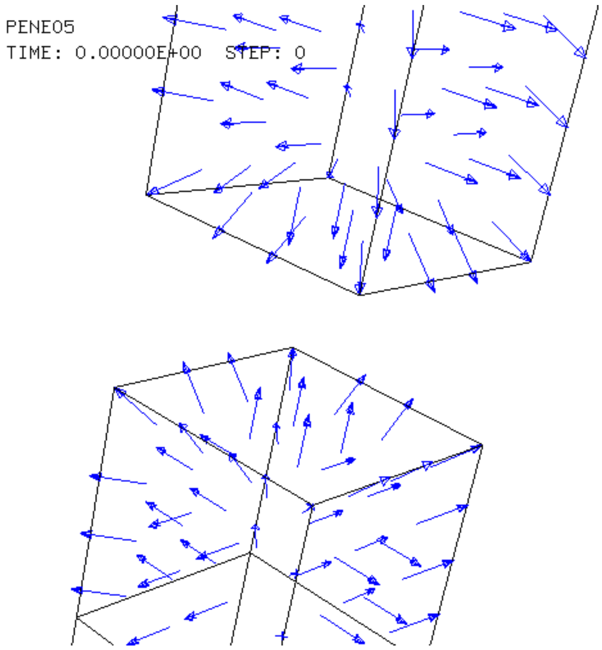


Figure 69 - ASNs in case PENE05.

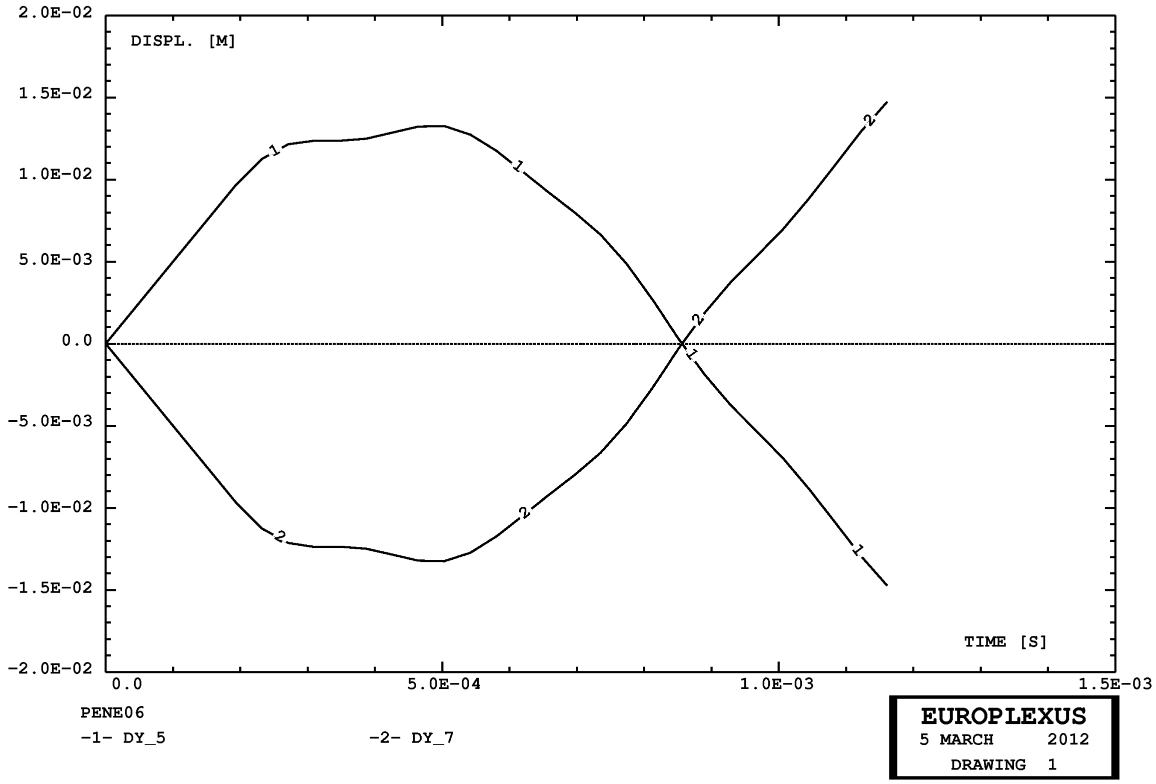


Figure 70 - Displacements in case PENE06.

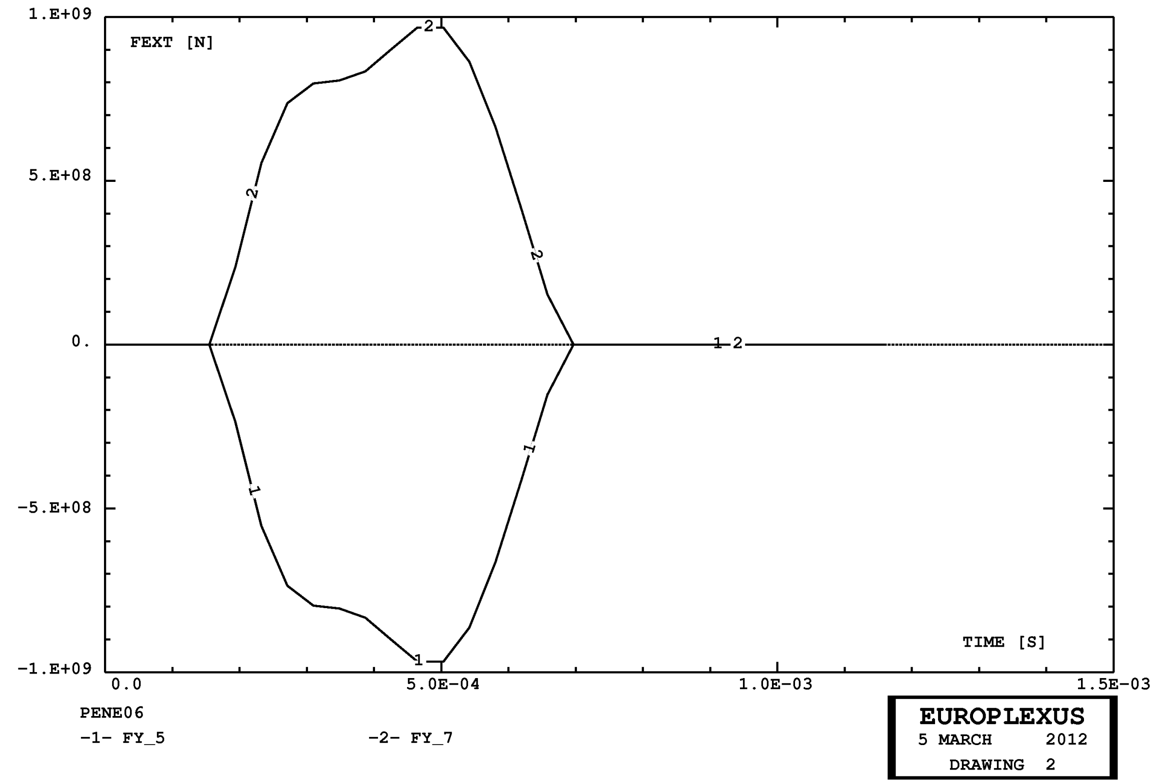


Figure 71 - Contact forces in case PENE06.

5.3.2 Impacts between material points

An advantage of the penalty formulation with respect to the Lagrange Multipliers formulation is that it allows to treat impacts between material points, i.e. between particles represented by one-node "elements" in the same way as for continuum or shell elements. The same is not true for the Lagrange Multipliers method (in its standard form), in which a difficulty would arise in modeling the rebound between the impacting particles.

An example is considered in Table 7:

Name	Mesh	Contact parameters	Description
PMAT01	2 PMAT	PENA DIAM 1.0 ASN	material point impact in 2D

Table 7 - Impact tests between material points.

PMAT01

Two particles hit each other at a certain initial velocity. The particles material is elastic, so after some interaction rebound occurs. Pinballs are associated to the particles, with a prescribed diameter (`DIAM`). These are parent (0-level) pinballs since the hierarchic pinball method cannot be applied to particles. A penalty method (`PENA`) is used. The `ASN` method is formally invoked: however, this has no influence in the present case since no ASN can be associated with a particle, and therefore the contact occurs always along the line joining the contacting particle centers.

A "thickness" (diameter, in this case) of 1.0 is assigned to the particles via the `COMP EPAI` directive. This is used for the visualization of the particles, but also for the calculation of their mass and critical time step. The latter two quantities are indeed computed because a material `LINE` (and not the more usual material `MASS`) is associated with the particles. The elastic properties of the material are used to compute the penalty forces during contact.

Note that the diameter of the pinballs associated with the particles is assigned independently in the `PINB DIAM` directive. Here the same value as the physical diameter of the particles (1.0 units) is chosen, which seems the only reasonable choice. However, be aware that the choice is left to the user and the code will use any value prescribed, without checking that it is equal to the value assigned to the particle diameter via the `COMP EPAI` directive.

Figure 72 shows the two particles (via the associated pinballs) during the contact phase, before rebound starts. Figures 73, 74 and 75 show the displacements, velocities and contact forces. As it can be observed, the rebound is elastic in that the rebound velocity is equal and opposite to the initial velocity, and the interaction is quite smooth (no oscillations).

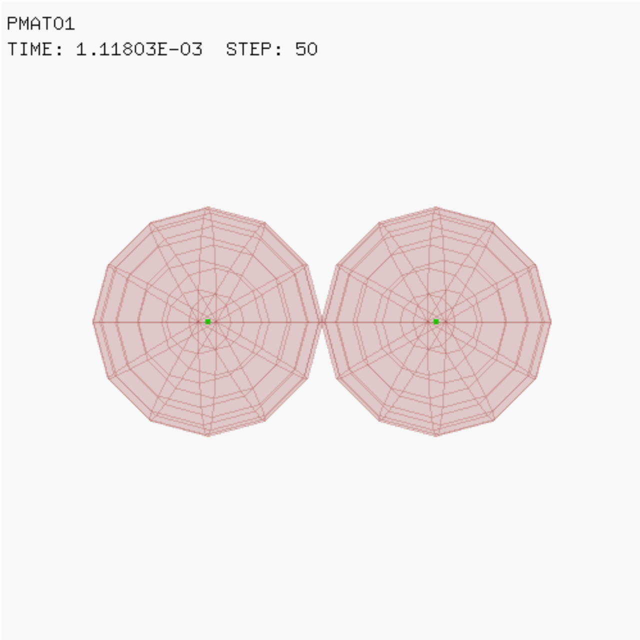


Figure 72 - Contacting particles in test PMAT01.

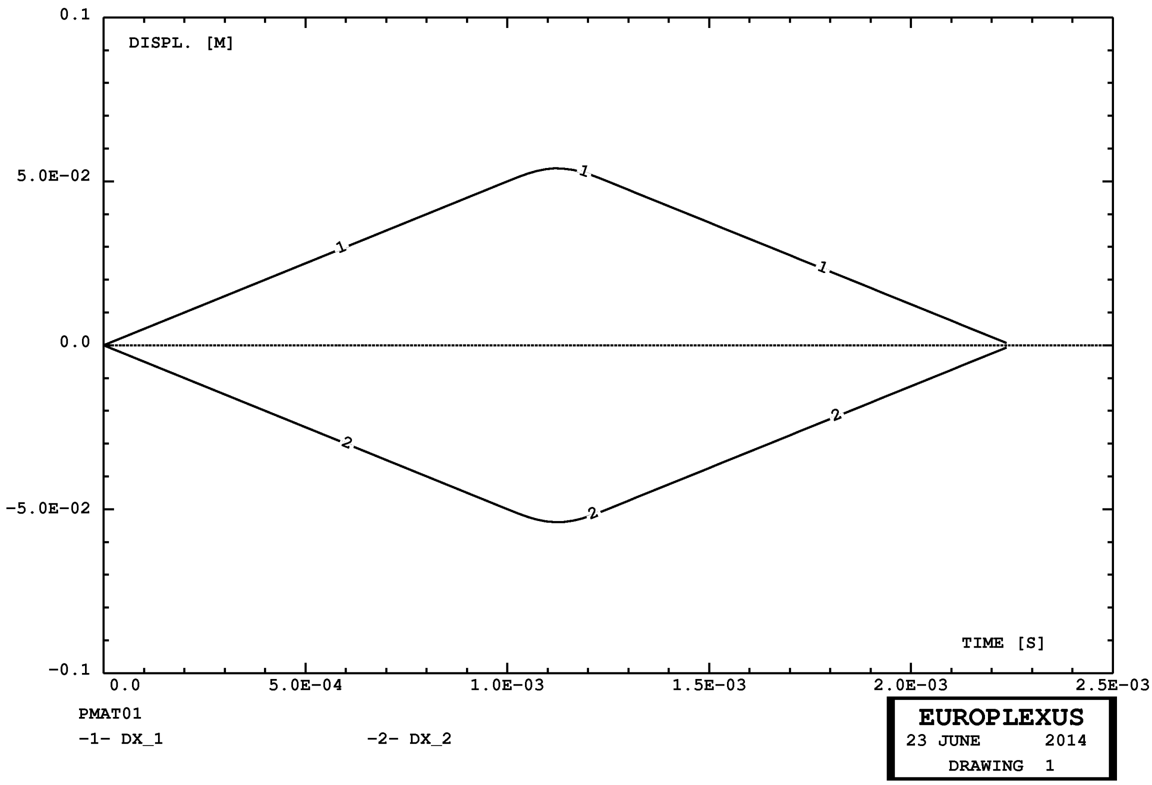


Figure 73 - Displacements in case PMAT01.

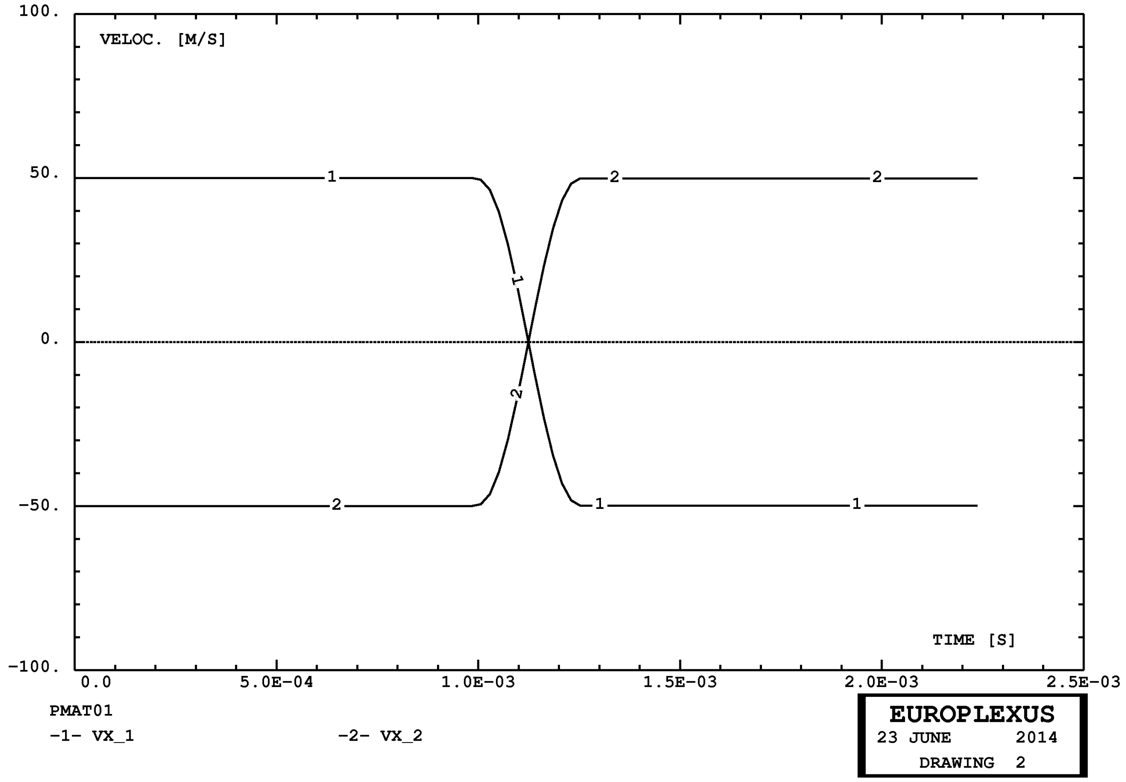


Figure 74 - Velocities in case PMAT01.

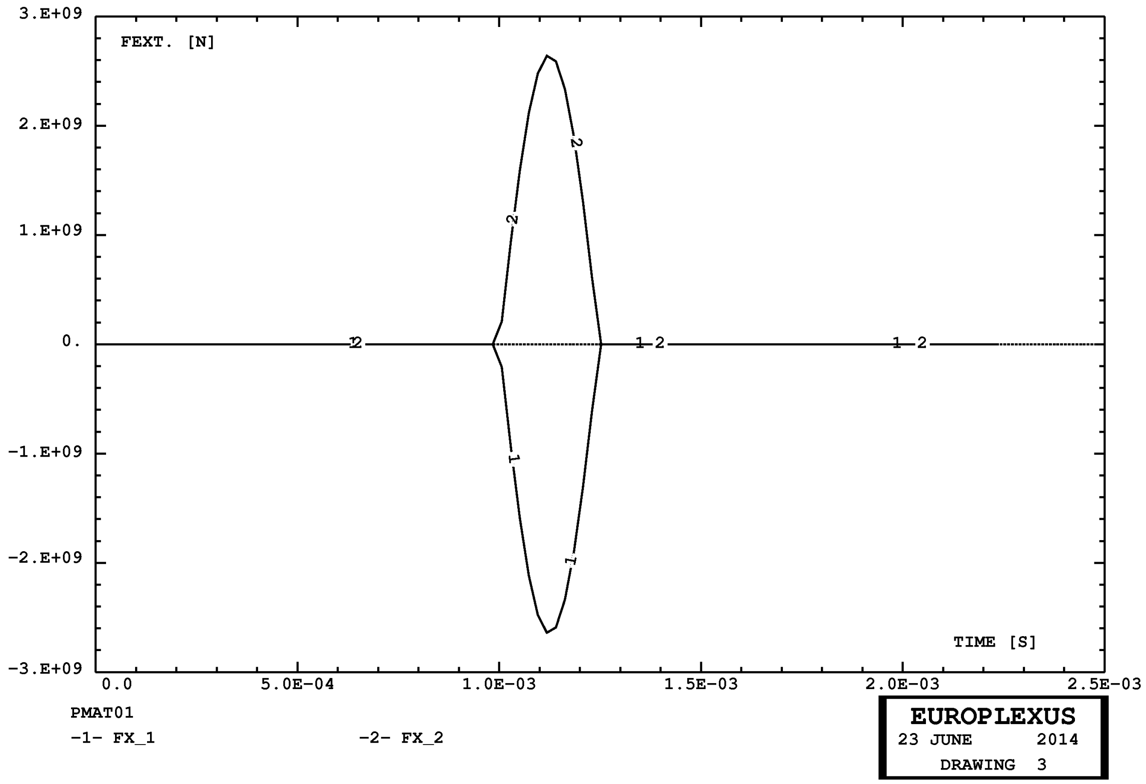


Figure 75 - Contact forces in case PMAT01.

5.3.3 Impacts between material points and other element types

We now consider impact between material points and other element types, namely shell elements.

Examples are considered in Table 7:

Name	Mesh	Contact parameters	Description
PMAT02	1 PMAT, 81 Q4GS	PENA ASN	material point impacts 3D plate
PMAT03	1 PMAT, 100 Q4GS	PENA ASN	same as 02 but finer mesh
PMAT04	1 PMAT, 361 Q4GS	PENA ASN	even finer mesh
PMAT05	1 PMAT, 400 Q4GS	PENA ASN	even finer mesh
PMAT06	1 PMAT, 361 Q4GS	LINK COUP ASN	same as 04 but LINK COUP
PMAT07	1 PMAT, 400 Q4GS	LINK COUP ASN	same as 05 but LINK COUP

Table 8 - Impact tests between material points and other element types.

PMAT02

A material particle hits a square plate at a certain initial velocity. Both the particle and the plate have linear elastic material. The plate is clamped along its contour (all displacements and all rotations blocked). The plate is discretized by a regular grid of 9×9 shell elements Q4GS. The `PENA` method with `ASN` is used to describe the contact.

Figure 76 shows an example of contact occurring during the test, namely the secondary contact when the deformed plate starts bouncing back and hits the particle. Figures 77, 78 and 79 show the displacement, velocity and contact force on the particle. Contact occurs in two phases: first the particle hits the plate, which deforms and detaches from the particle. Then the plate bounces back and hits the particle. Contact is quite smooth, as already observed in the previous example.

PMAT03, PMAT04 and PMAT05

These tests are repetitions of case PMAT02 using finer and finer meshes for the plate, up to 400 elements, in order to check the convergence of the numerical solution to a stabilized result. The reason for using meshes differing by just one element along each spatial direction is that when the element number is odd the impact occurs at the centre of an element (face pinball) while when the number is even the impact occurs exactly at one node of the plate mesh. Results should not be sensitive to such details.

Figures 80, 81 and 82 compare solutions PMAT02 and PMAT03 (9 and 10 shell elements respectively), showing some significant differences, so that finer meshes are needed.

Figures 83, 84 and 85 compare solutions PMAT04 and PMAT05 (19 and 20 shell elements respectively), showing better convergence although some differences remain especially in the contact forces.

PMAT06 and PMAT05

These tests are repetitions of cases PMAT04 (19 elements) and PMAT05 (20 elements), respectively, by using Lagrange Multipliers (coupled links) instead of penalty method to describe contact. The ASN algorithm is kept and is used in combination with the Lagrange Multipliers method.

Results of these two calculations are compared with case PMAT05 in Figures 86, 87 and 88. While solution PMAT06 is relatively similar to PMAT05, solution PMAT07 has a quite different behavior. Contact forces tend to act over much longer periods in the solutions with Lagrange Multipliers, compared with those with penalty. This might indicate a problem in treating the rebound (which is necessary with the Lagrange Multipliers while it is redundant with the penalty method).

This problem will be better investigated in a subsequent Section, dealing with sliding-like contact between continuum bodies.

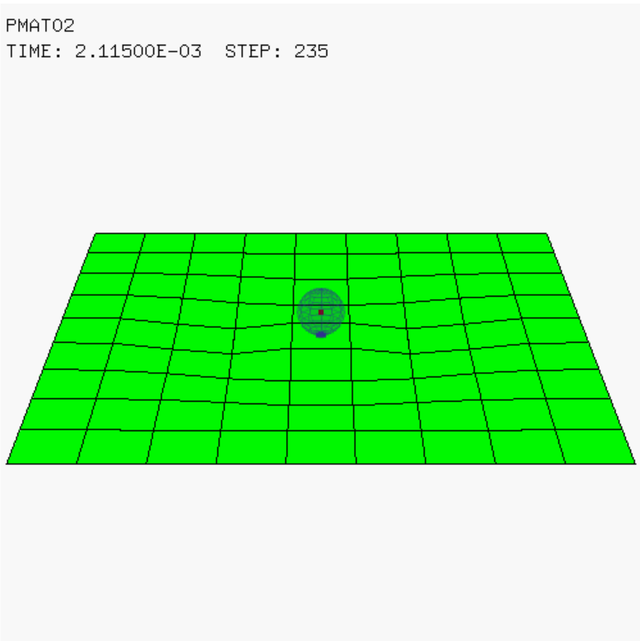


Figure 76 - Contact in test PMAT02.

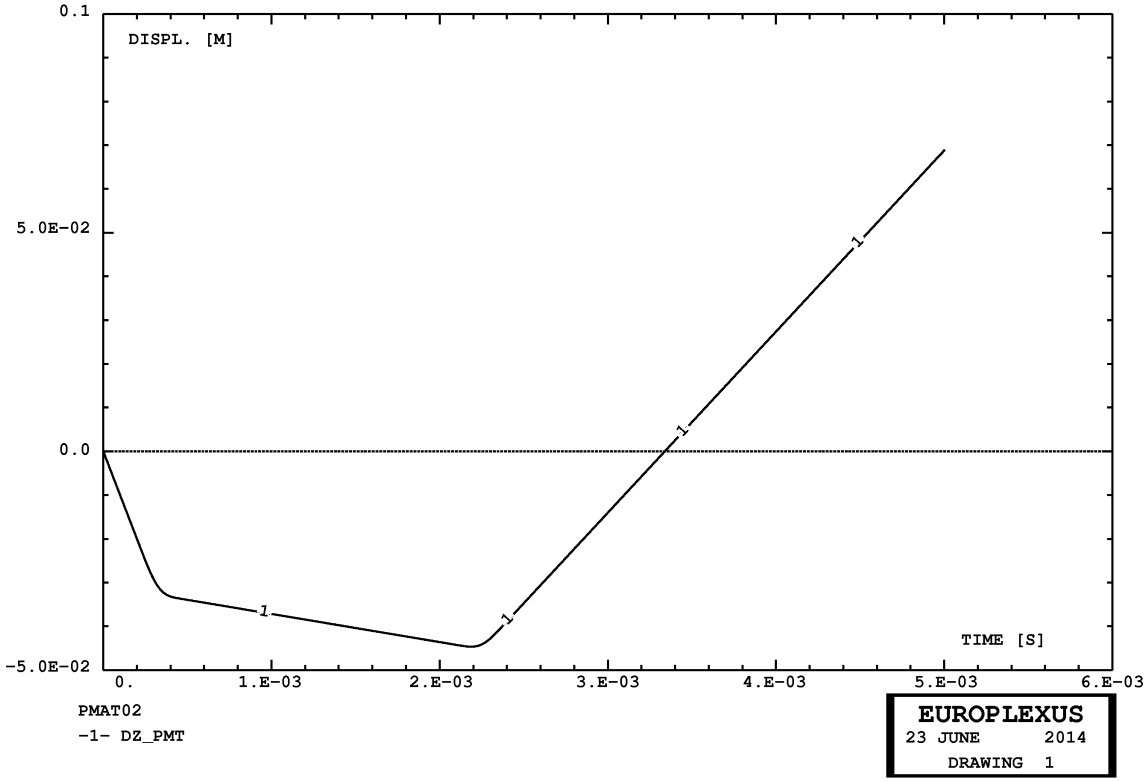


Figure 77 - Particle displacement in case PMAT02.

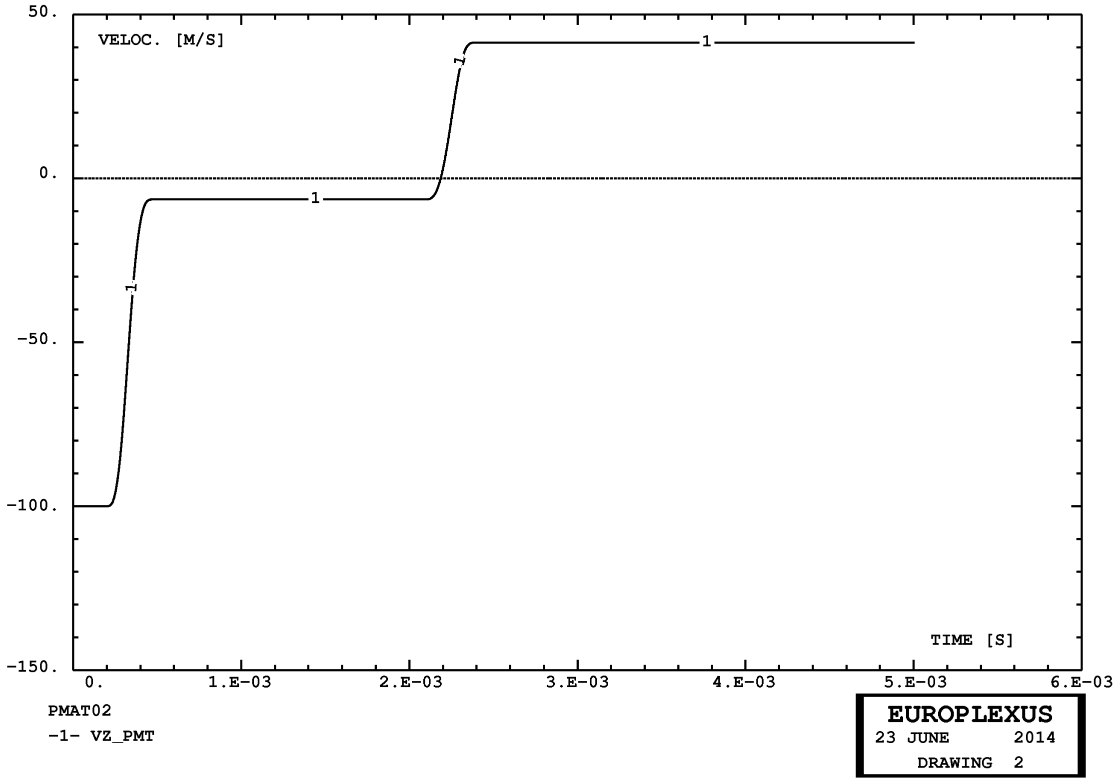


Figure 78 - Particle velocity in case PMAT02.

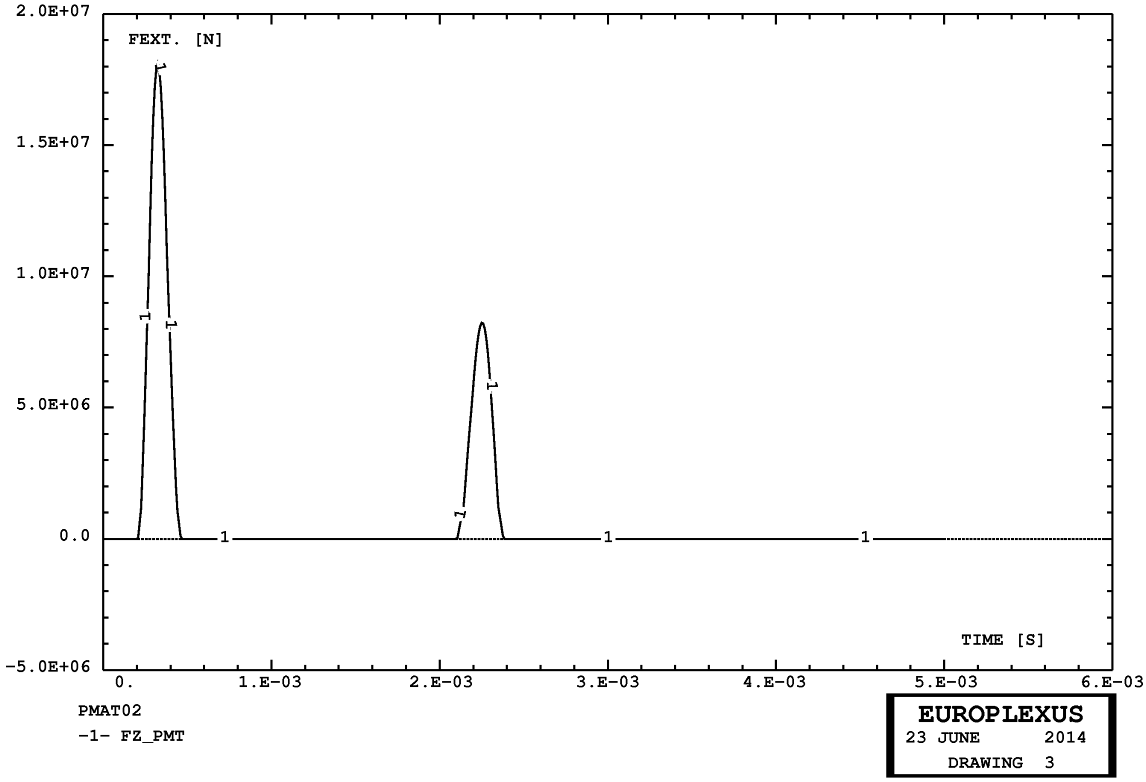


Figure 79 - Particle contact force in case PMAT02.

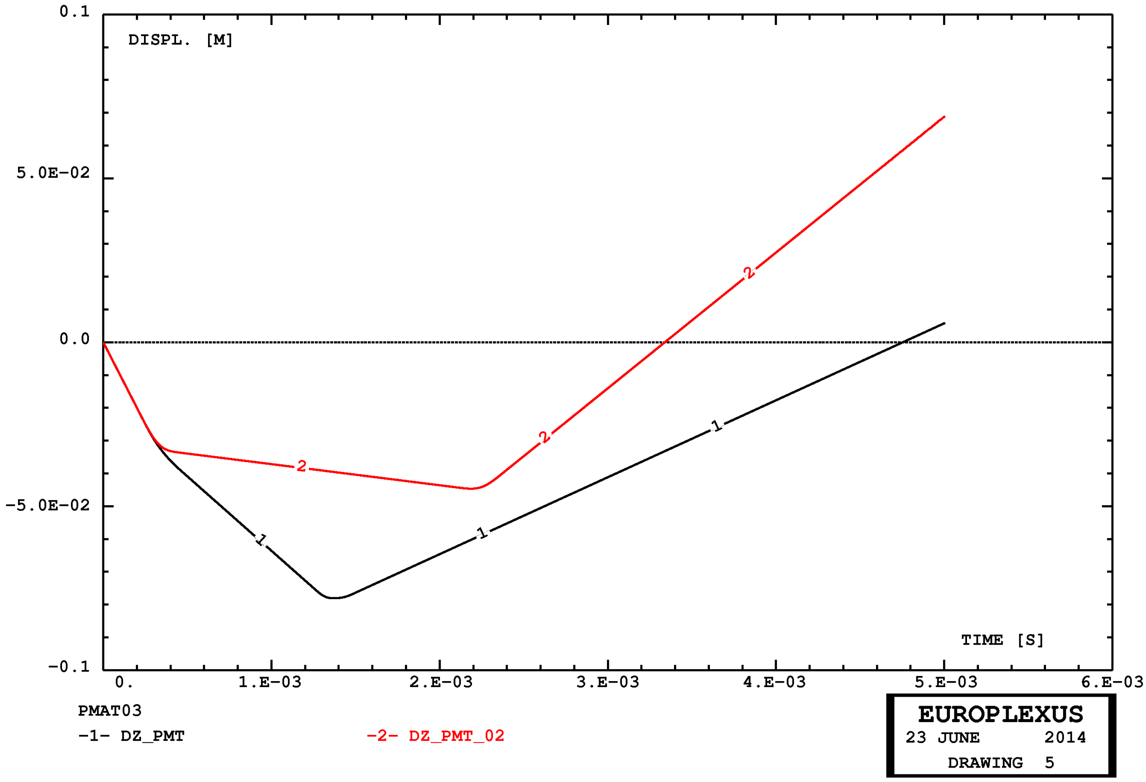


Figure 80 - Particle displacement in cases PMAT02 and PMAT03.

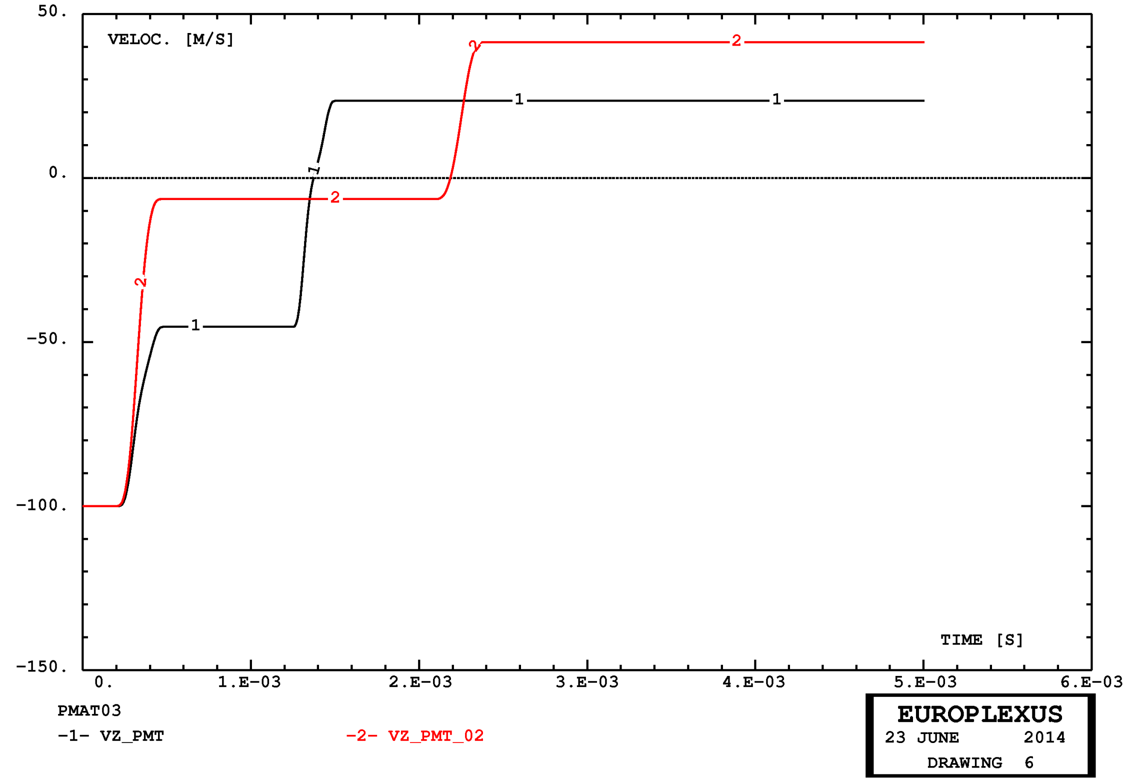


Figure 81 - Particle velocity in cases PMAT02 and PMAT03.

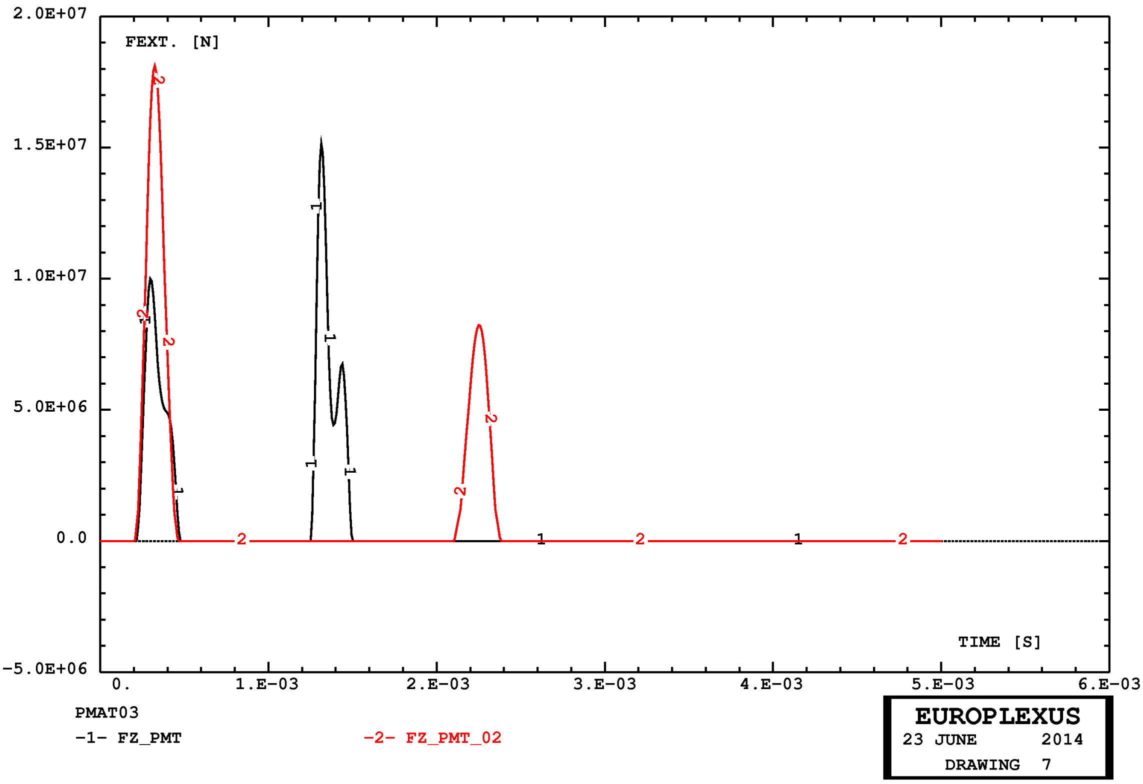


Figure 82 - Particle contact force in cases PMAT02 and PMAT03.

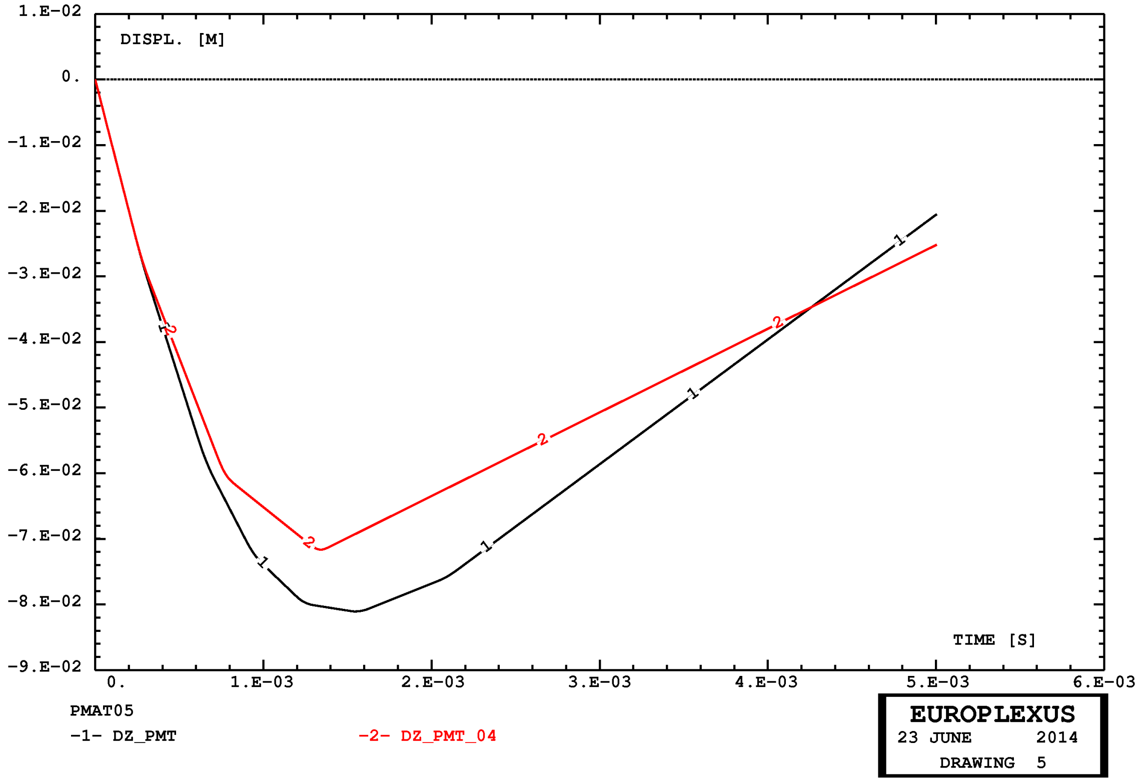


Figure 83 - Particle displacement in cases PMAT04 and PMAT05.

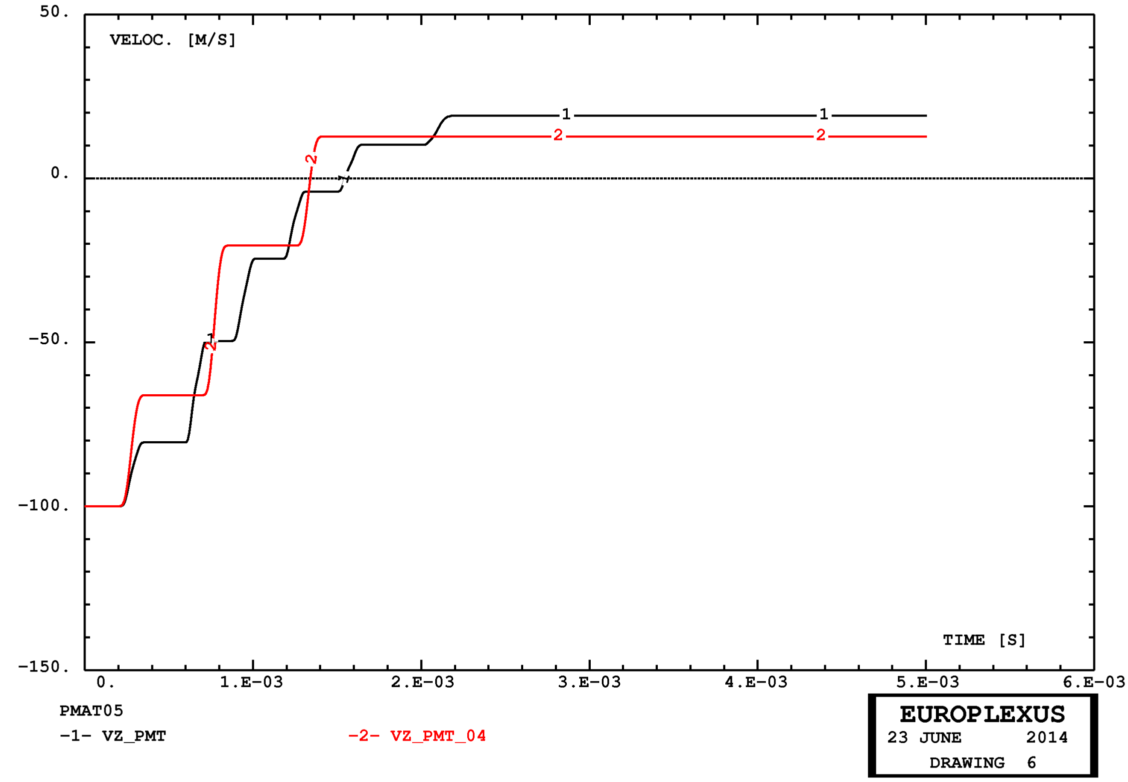


Figure 84 - Particle velocity in cases PMAT04 and PMAT05.

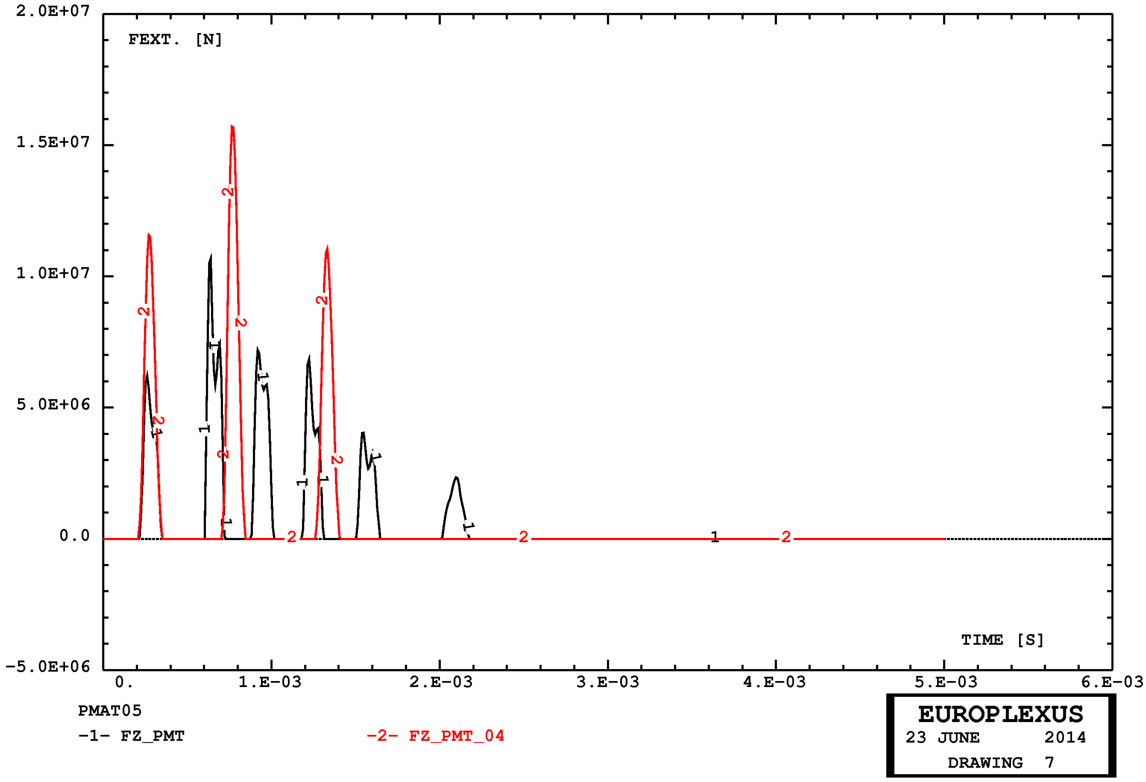


Figure 85 - Particle contact force in cases PMAT04 and PMAT05.

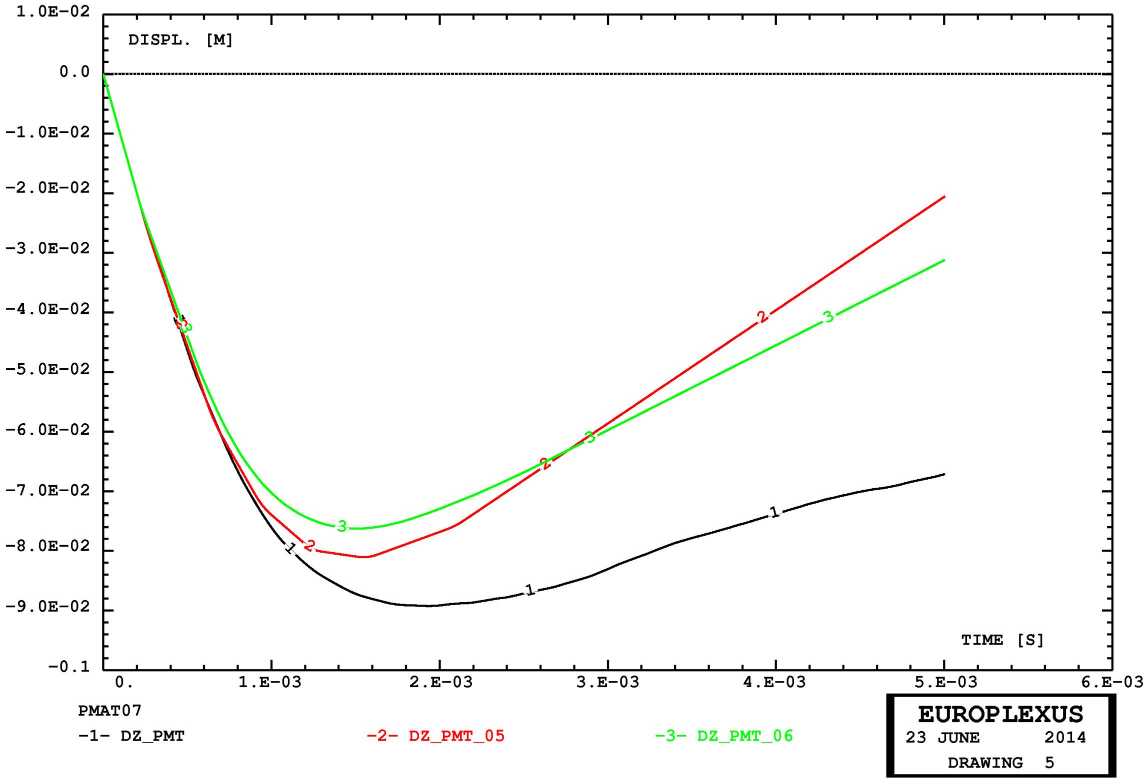


Figure 86 - Particle displacement in cases PMAT05, PMAT06 and PMAT07.

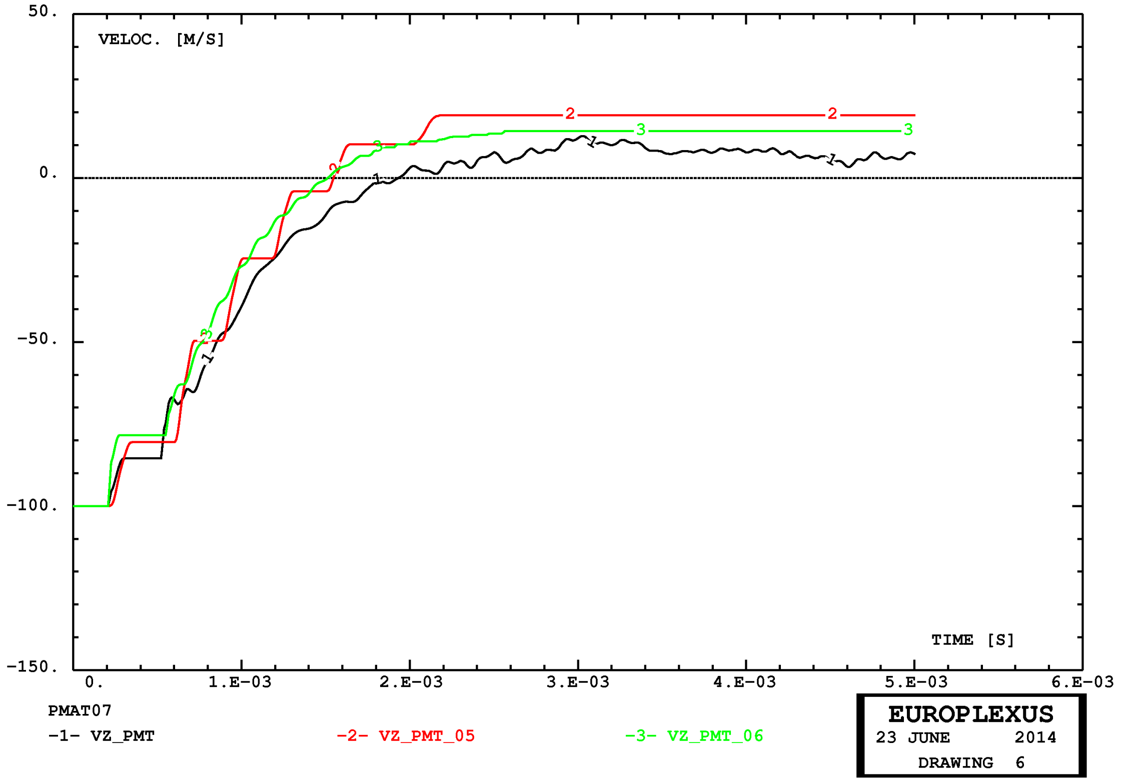


Figure 87 - Particle velocity in cases PMAT05, PMAT06 and PMAT07.

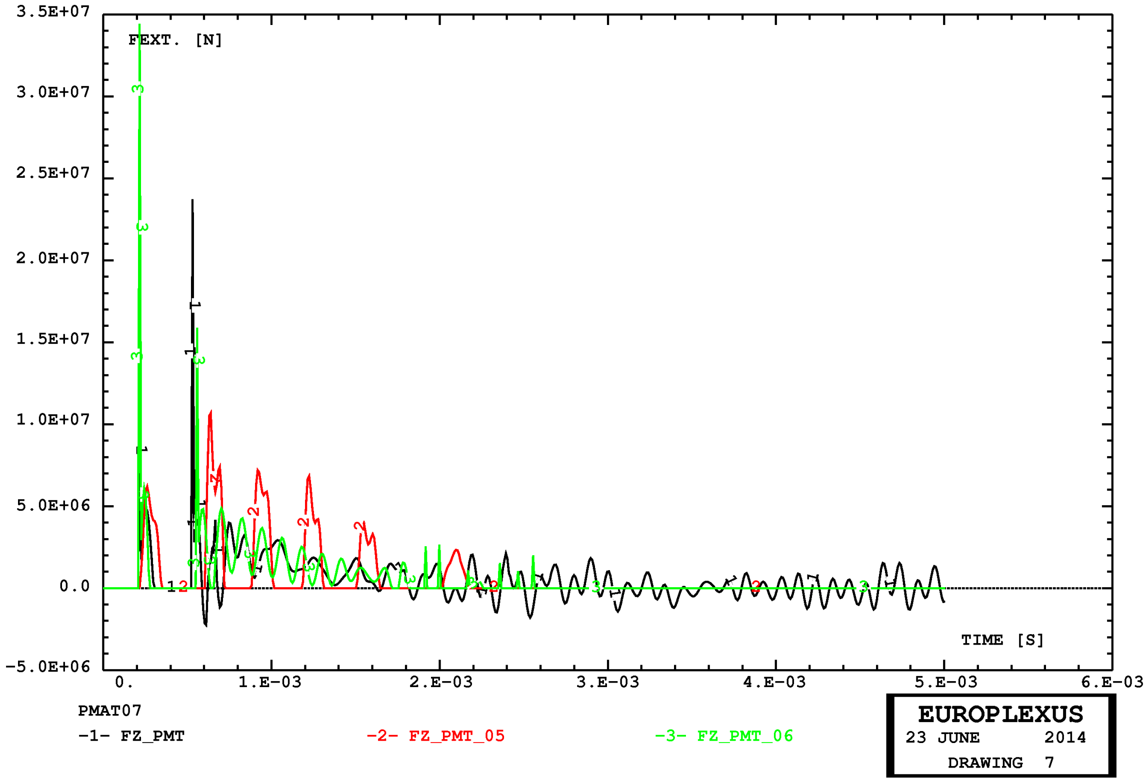


Figure 88 - Particle contact force in cases PMAT05, PMAT06 and PMAT07.

5.3.4 Impacts of “sliding” type between continuum bodies

Finally, we consider impact (or rather contact) of “sliding” type between continuum elements. This type of phenomenon is less violent than impact, but presents some particularities that make it a challenging test for the proposed algorithms.

The problem consists of two blocks of metal, a lower block of rectangular shape which is fixed at its base. and an upper block of square shape initially lying slightly above the first block and with an initial velocity causing a slight impact against (and a lot of sliding along) the fixed block. The horizontal component of the velocity is 100 m/s while the vertical component is -10 m/s. The lower block is discretized by $10 \times 2 = 20$ quadrilateral elements CAR1, while the upper block is discretized by $2 \times 2 = 4$ CAR1. The material is linear elastic with steel-like characteristics.

The executed calculations are listed in Table 7:

Name	Mesh	Contact parameters	Description
SLID01	24 CAR1	PENA MLEV 4	Penalty solution *without* ASN
SLID02	24 CAR1	PENA MLEV 4 ASN	Penalty solution *with* ASN
SLID03	24 CAR1	LINK COUP MLEV 4	Lagrange mult. solution *without* ASN
SLID04	24 CAR1	LINK COUP MLEV 4 ASN	Lagrange mult. solution *with* ASN
SLID05	2 CAR1	LINK COUP MLEV 4	Lagrange mult. solution *without* ASN
SLID06	2 CAR1	LINK COUP MLEV 4 ASN	Lagrange mult. solution *with* ASN
SLID07	2 CAR1	LINK COUP MLEV 3 ASN	Lagrange mult. solution *with* ASN
SLID08	2 CAR1	LINK COUP MLEV 2 ASN	Lagrange mult. solution *with* ASN
SLID09	2 CAR1	LINK COUP MLEV 1 ASN	Lagrange mult. solution *with* ASN
SLID10	2 CAR1	LINK COUP MLEV 0 ASN	Lagrange mult. solution *with* ASN

Table 9 - Sliding-type contact between continuum elements.

SLID01

This solution uses the penalty (decoupled) approach to contact, without the ASN algorithm. Two time instants of the solution are presented in Figure 89: the initial configuration and the instant of first contact between the two blocks (step 85).

Figure 90 shows the vertical displacement of the upper block, indicating that rebound (although with some elastic oscillations) occurs regularly. The velocity components of the upper block are shown in Figure 91. The horizontal component should be conserved since there is no friction and the two contacting surfaces are parallel. A slight loss takes place due to the irregular (non-flat) shape of the pinballs: recall that in this case the ASN is not used so the contact occurs in a slightly non-vertical direction. The vertical component indicates rebound with some oscillations. Finally, Figure 92 shows the contact force components: there is a small but non-negligible horizontal component (which should not be there as already mentioned) and this is the cause for dissipation of horizontal velocity.

SLID02

This solution is similar to the previous one but the ASN is activated. The contact direction should therefore much closer to the ideal one, which is perfectly vertical (apart some distortions induced by the elastic deformation of the bodies).

Figures 93, 94 and 95 confirm that the solution is better than the previous one: the horizontal velocity stays almost perfectly constant (see also finer comparison in Figure 96 between the two solutions) and the horizontal contact force is practically zero.

SLID03

This solution is similar to SLID01 (no ASN) but uses the Lagrange Multiplier algorithm (`LINK COUP`) instead of the penalty approach. The solution, shown in Figures 97, 98 and 99, is very similar to SLID01: slight loss of horizontal velocity due to a spurious (but small) horizontal contact force. The rebound appears correct.

SLID04

This solution is similar to SLID02 (ASN) but uses the Lagrange Multiplier algorithm (`LINK COUP`) instead of the penalty approach. It would be expected that this solution be better than SLID03 and similar to SLID02 obtained with the penalty method.

However, it is not so: as shown in Figure 100, at a certain moment of the solution the upper block seems to “stick” onto the lower one and starts rotating instead of sliding smoothly. This is confirmed from Figures 101, 102 and 103. The horizontal velocity is completely lost as an enormous horizontal component of the contact force suddenly appears.

The following tests are an attempt at understanding the problem. An obvious candidate for the observed mis-functioning is the rebound model, i.e. the so-called *a-priori* rebound model described in Section 8.2 of reference [13]. Recall in fact that a rebound model is necessary with the LM version of the contact model by pinballs (unlike in the penalty formulation) and that by default the a-priori rebound model is activated.

Another possible source of trouble is the fact that, as shown in Section 7 of reference [13], the Lagrangian Multipliers approach suffers from redundant constraints which render the links matrix singular: for the configuration of the simplified debugging test SLID05 and following ones (see next) the matrix is singular already from a hierarchy level of 2, as shown in Section 7.1 (pages 77-79) of [13].

The further debugging tests are done on a simplified version of the sliding contact problem: only one element is used for the lower block and one element for the upper block, in order to simplify the checks.

SLID05

This solution is similar to SLID03 (`LINK COUP` without ASN) but uses the reduced model with only two elements. The solution is acceptable, see Figure 105. The contact forces are not vertical, but this is normal since ASN is not activated in this case. Contact forces are repulsive (or zero), as suggested by the physics of the problem, so the a-priori rebound algorithm seems to work in this case.

SLID06

This solution is similar to SLID05 (2-element mesh) but uses the ASN (like in case SLID04). The solution is obviously unacceptable, as appears from Figure 105. Until step 4 the contact forces are repulsive, but from step 5 onwards they become <u>attractive</u>, which is clearly unphysical. The contact forces are vertical as expected, thanks to the ASN method, but the sign is obviously wrong.

Figure 106 shows the contact configurations in steps 0 to 7 for this calculation.

SLID07 to SLID10

These solutions are repetitions of case SLID06 (which uses `MLEV 4`) but with smaller hierarchy levels equal to 3, 2, 1 and 0, respectively. The scope is to see whether the inversion of contact forces occurs also at lower hierarchy levels.

Figure 107 shows the contacts in case SLID07 (MLEV 3). Contact force inversion occurs at step 4 instead of 5, but for the rest the solution is similar to SLID06. The same happens in SLID08 (MLEV 2) as shown in Figure 108.

The result of case SLID09 (MLEV 1) is strange and is shown in Figure 109. At this level there should be no constraint redundancy. However, strangely the contact forces become non-vertical from step 1 onwards, like if the ASN condition would not be respected. The reason for this can be seen in Figure 110: at step 1 the resulting ASN is indeed not vertical due to the particular contact configuration. It seems difficult to avoid such problems in general.

Finally, the case SLID10 (MLEV 0) is a limit case, as expected. The contact occurs only at step 0, (with vertical forces) which is sufficient to completely block the penetration. From step 1 onwards no contact forces occur and perfect sliding takes place, see Figure 111.

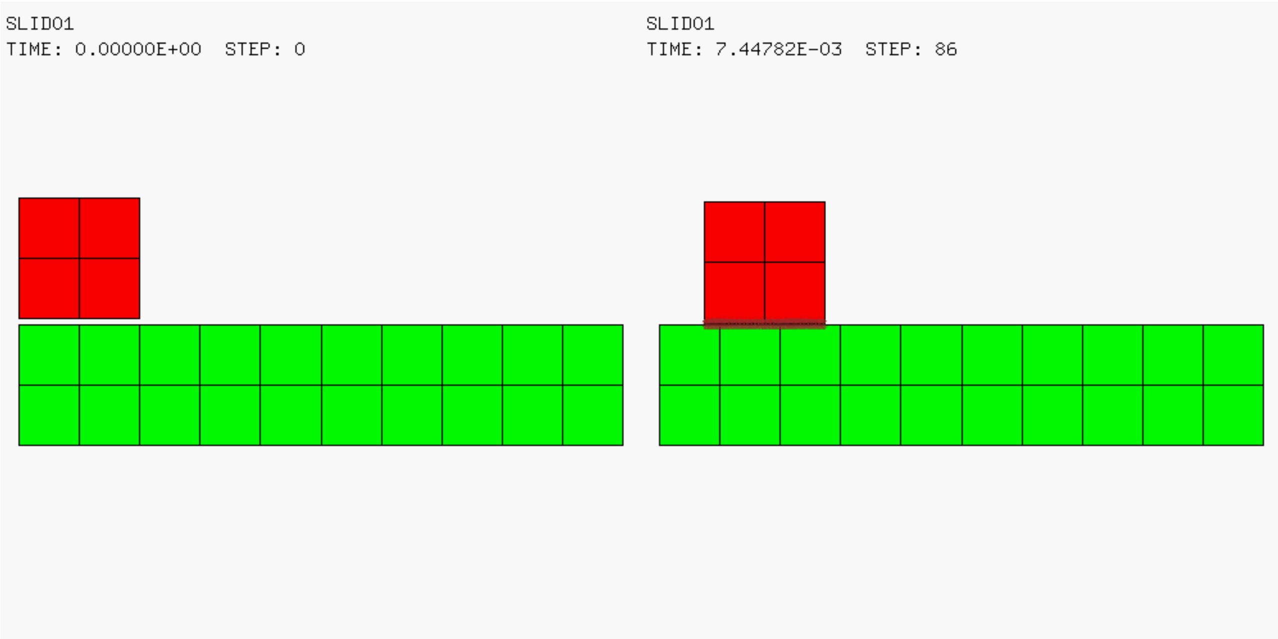


Figure 89 - Sliding contact test SLID01.

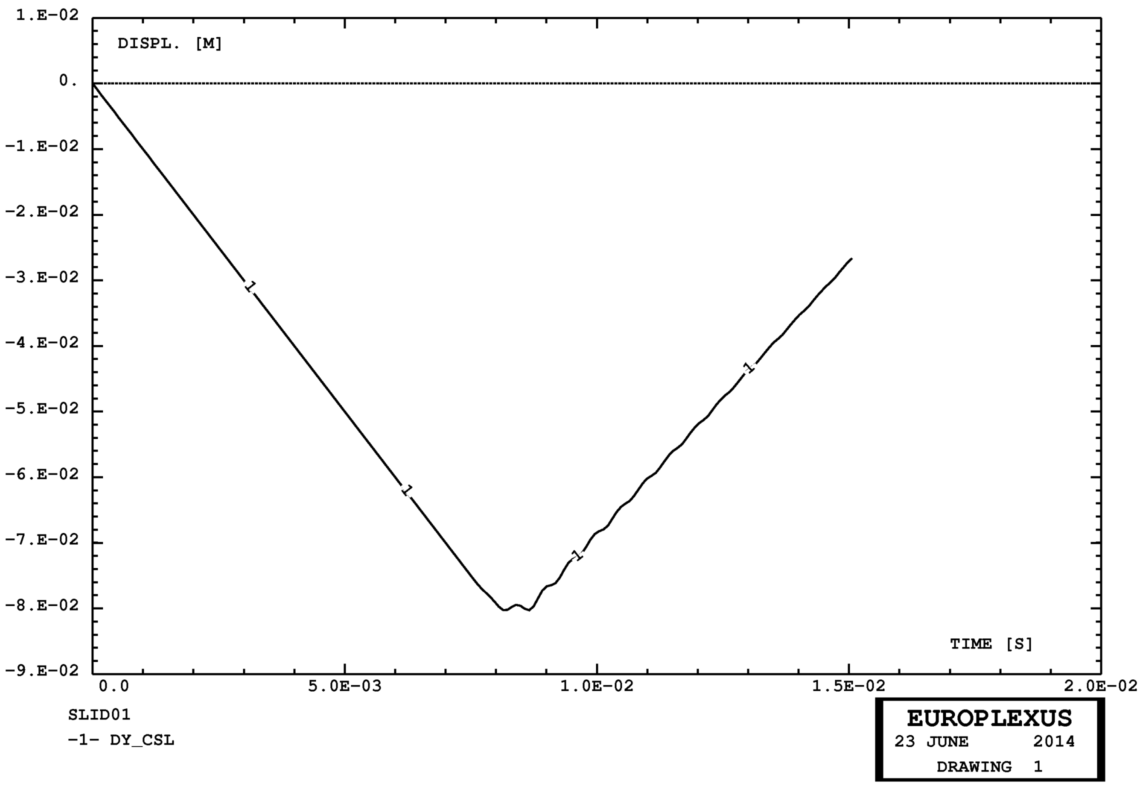


Figure 90 - Displacement of the upper block center in case SLID01.

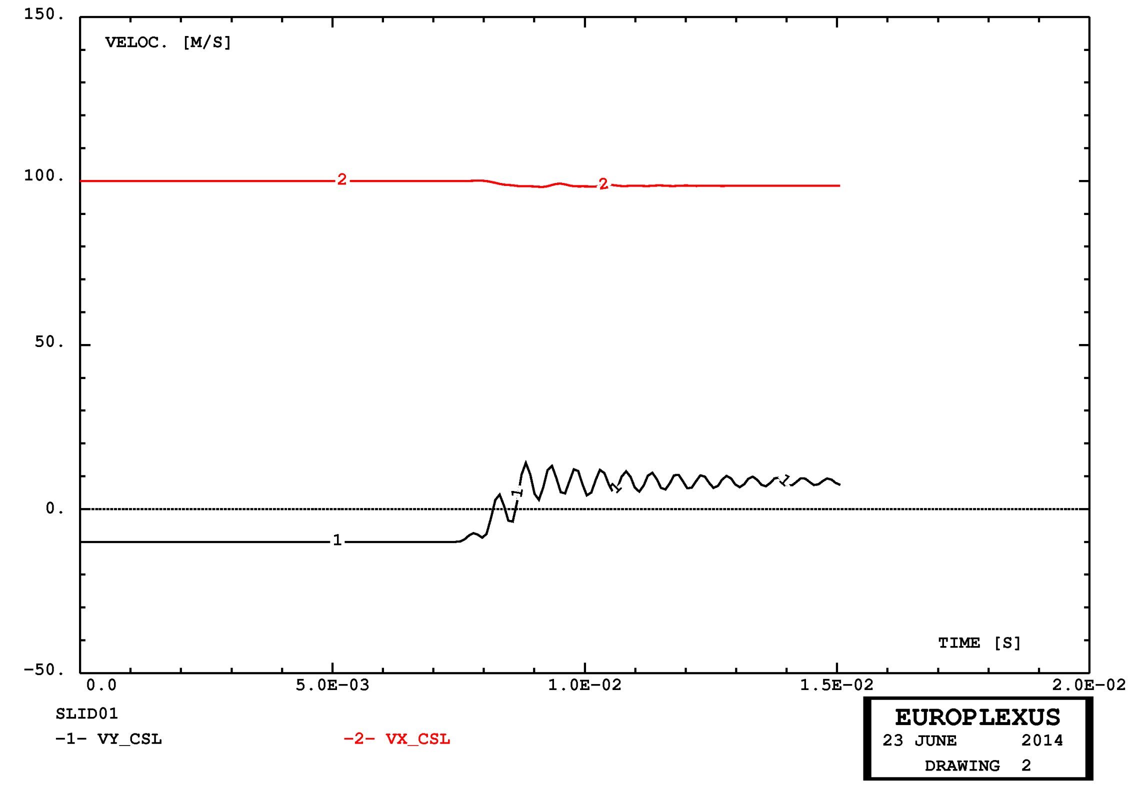


Figure 91 - Velocity components of the upper block center in case SLID01.

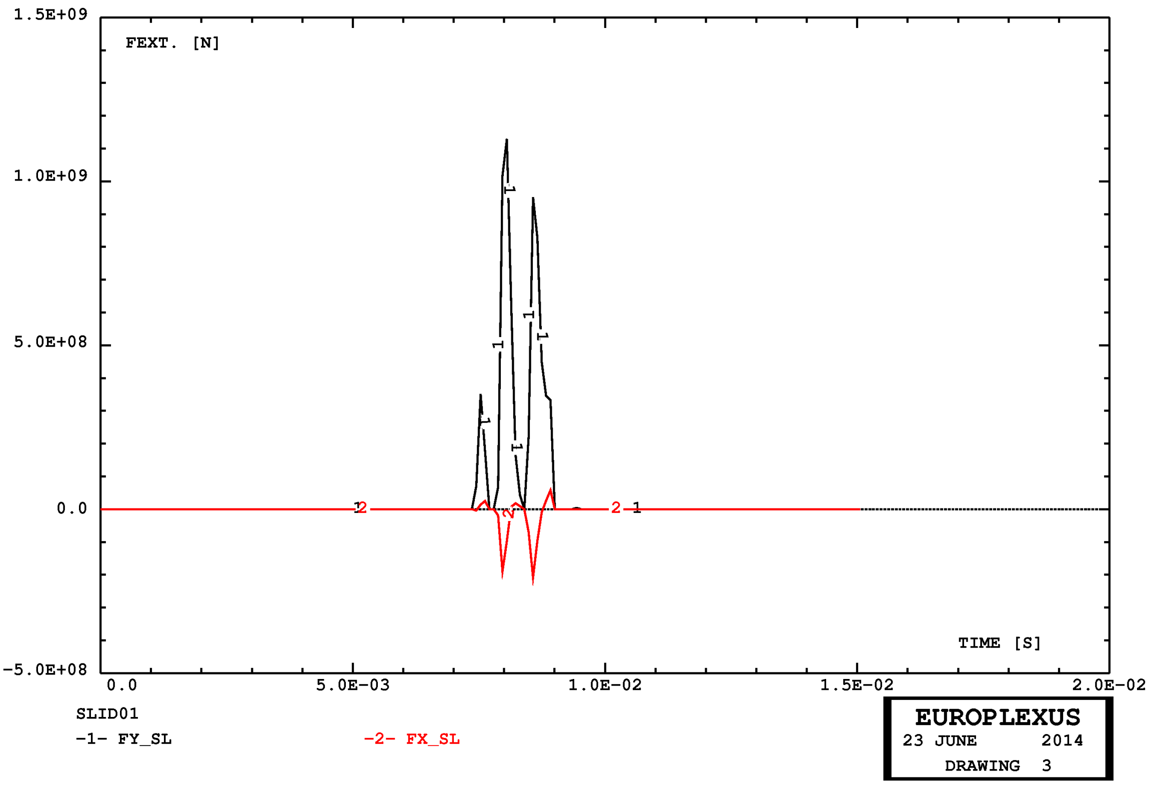


Figure 92 - Contact force components of the upper block center in case SLID01.

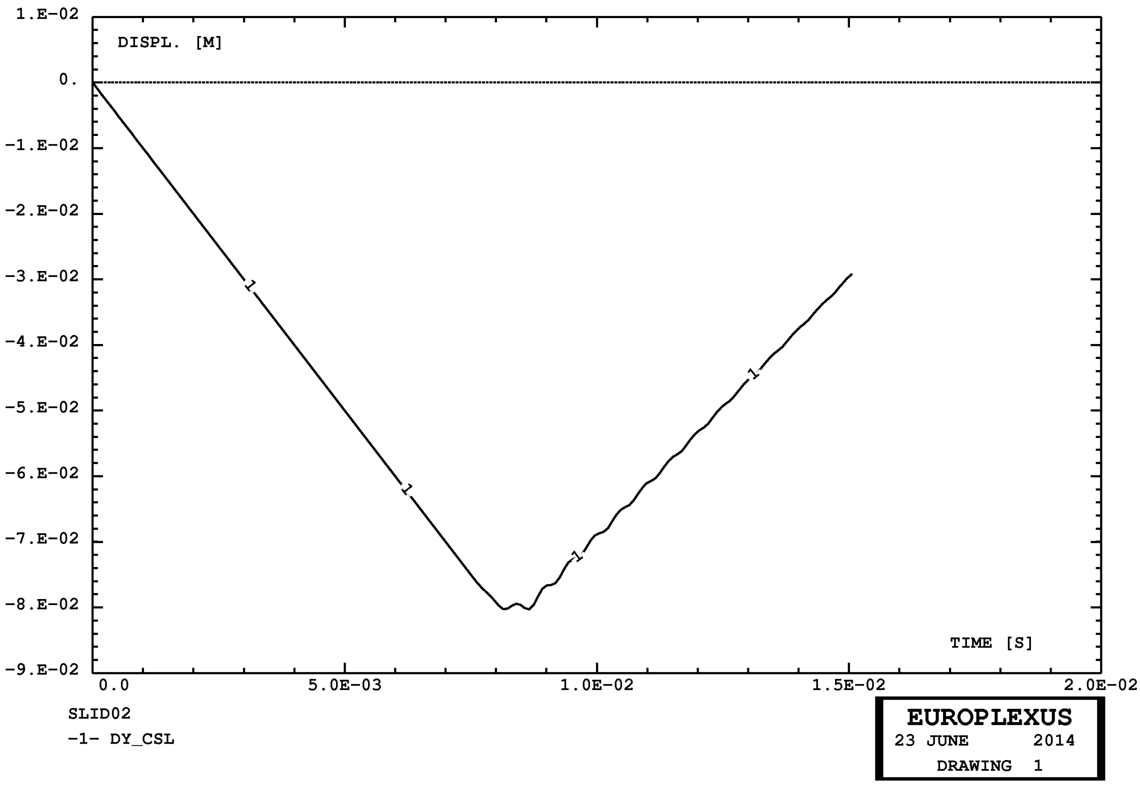


Figure 93 - Displacement of the upper block center in case SLID02.

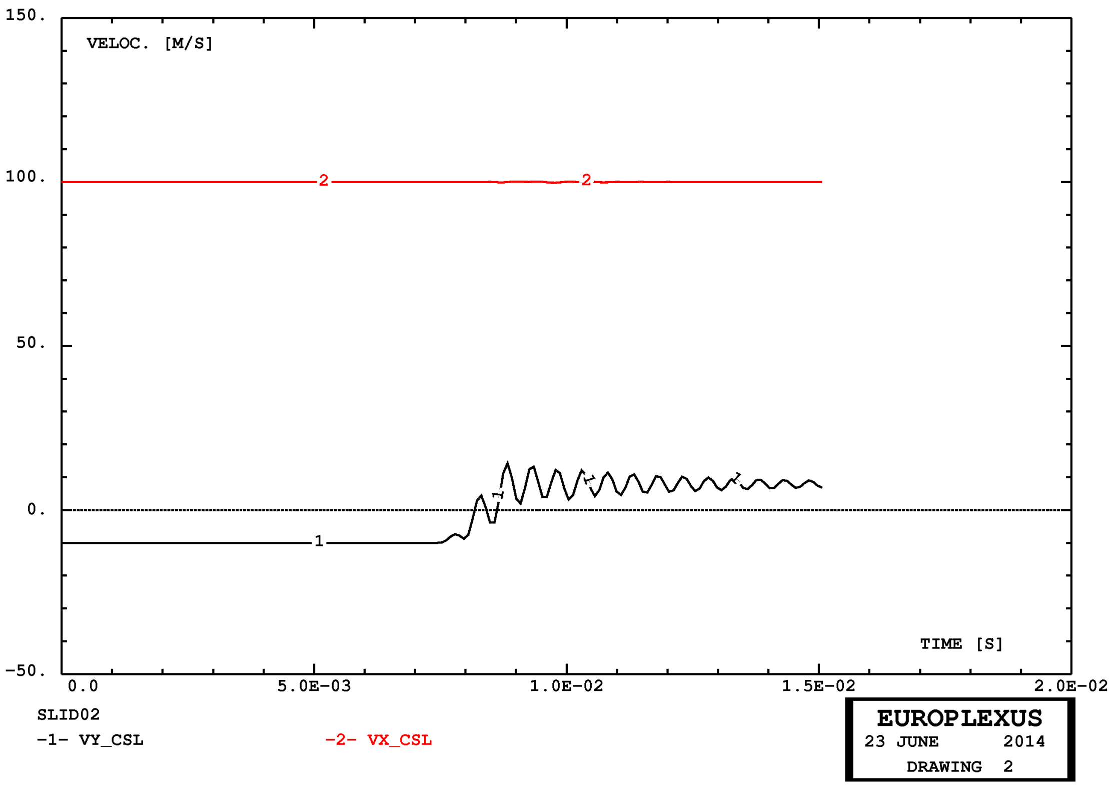


Figure 94 - Velocity components of the upper block center in case SLID02.

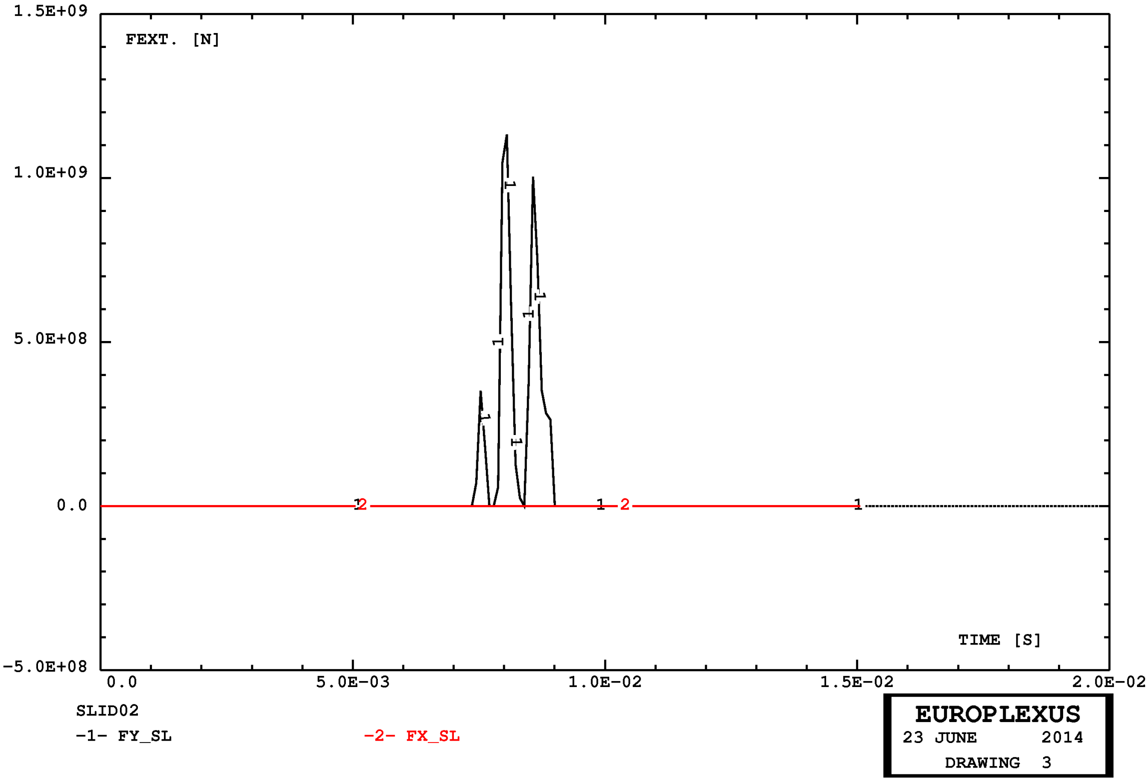


Figure 95 - Contact force components of the upper block center in case SLID02.

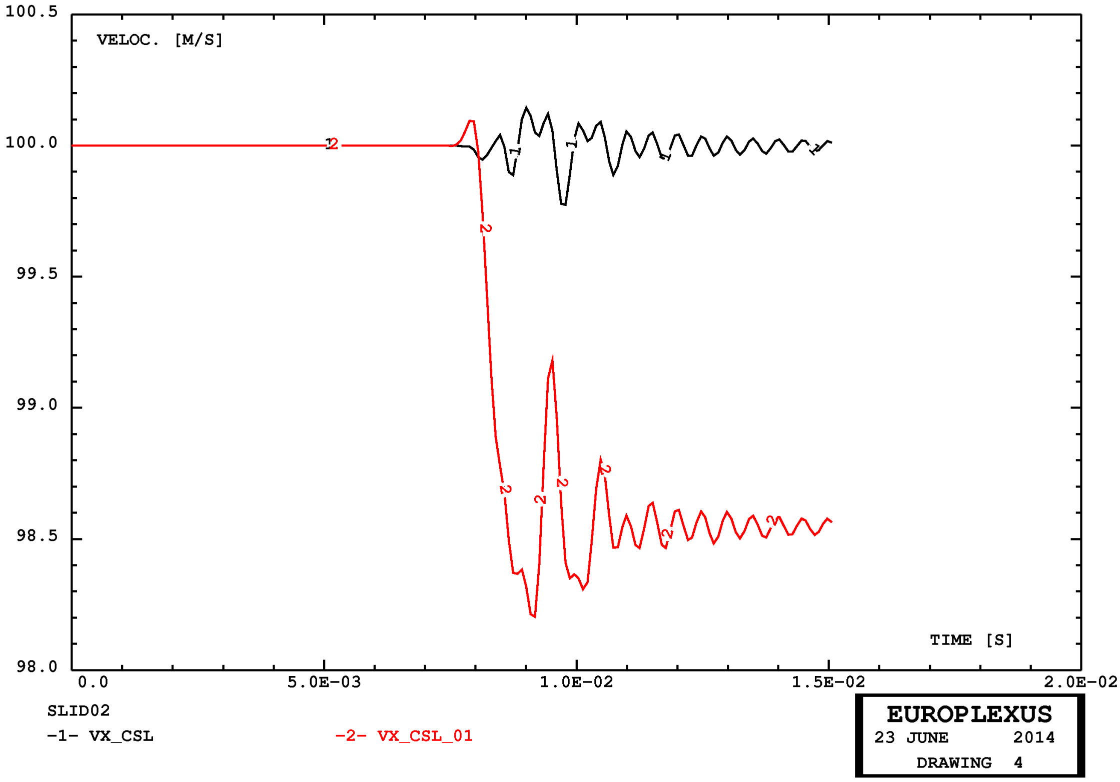


Figure 96 - Horizontal velocity components of the upper block center in cases SLID01 and SLI02.

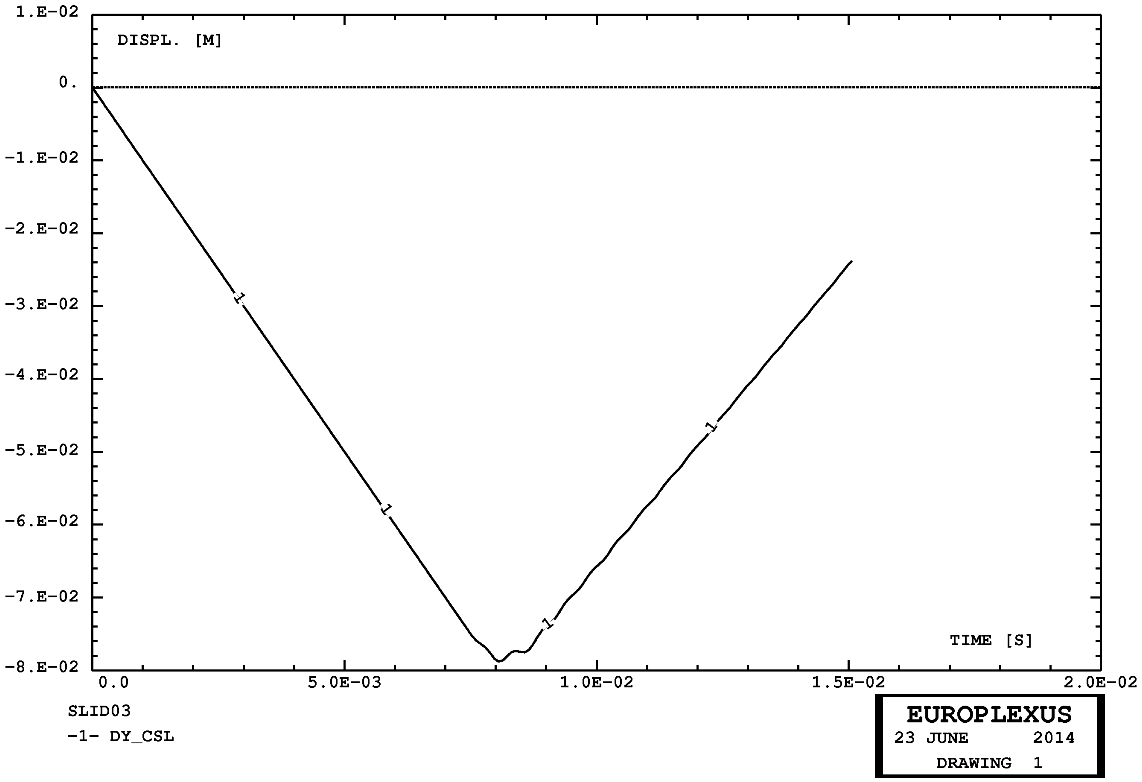


Figure 97 - Displacement of the upper block center in case SLID03.

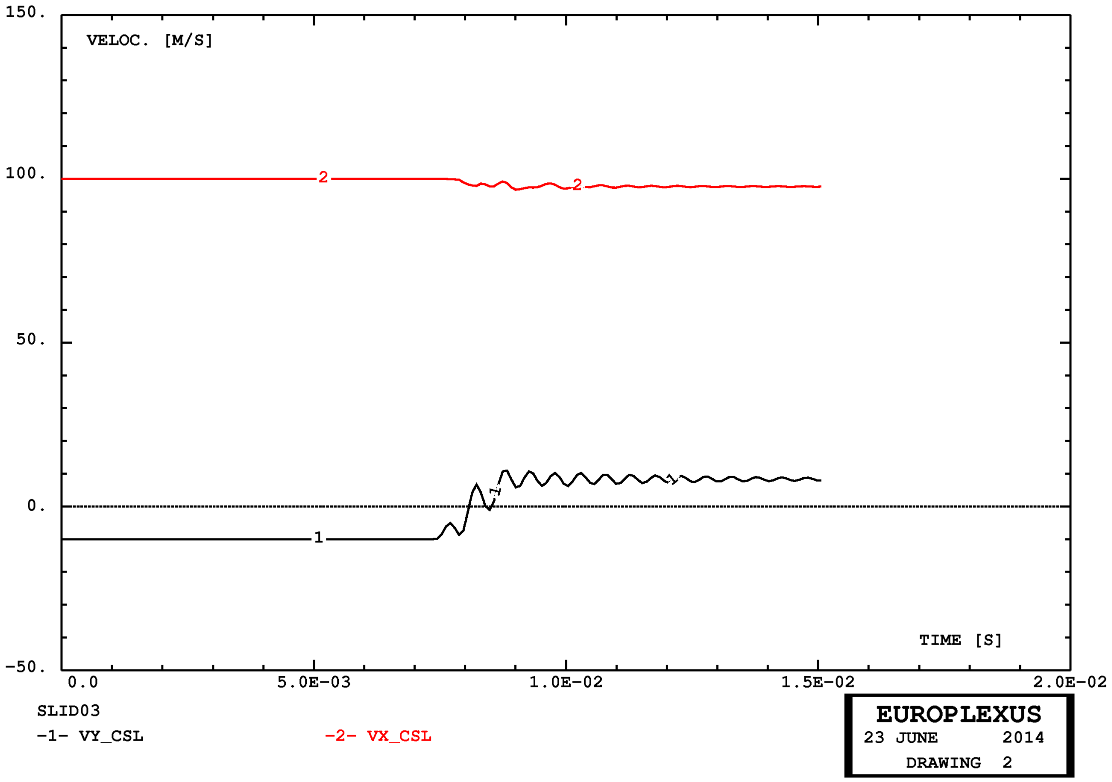


Figure 98 - Velocity components of the upper block center in case SLID03.

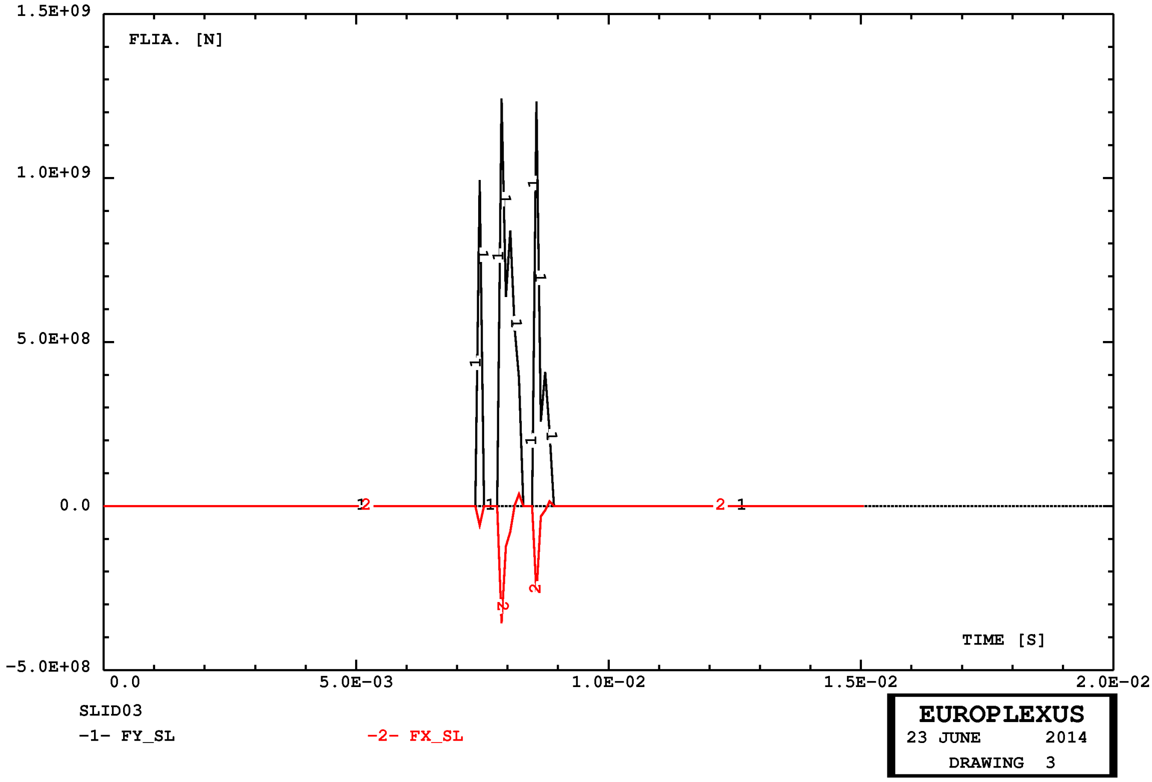


Figure 99 - Contact force components of the upper block center in case SLID03.

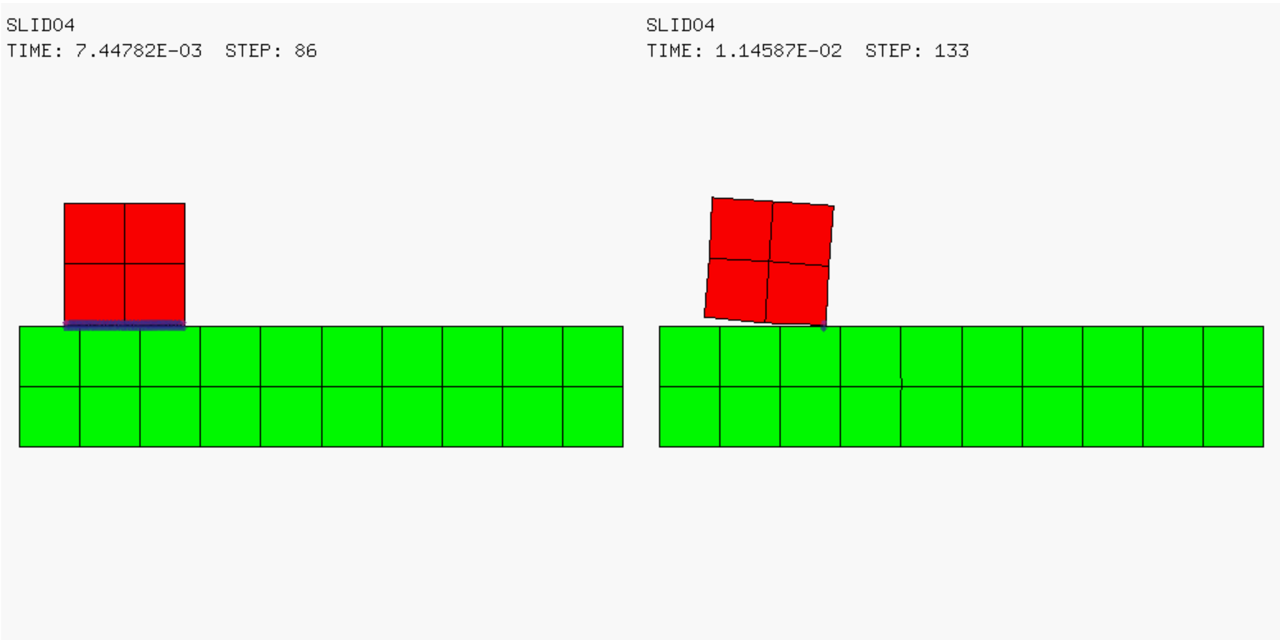


Figure 100 - Sliding contact test SLID04.

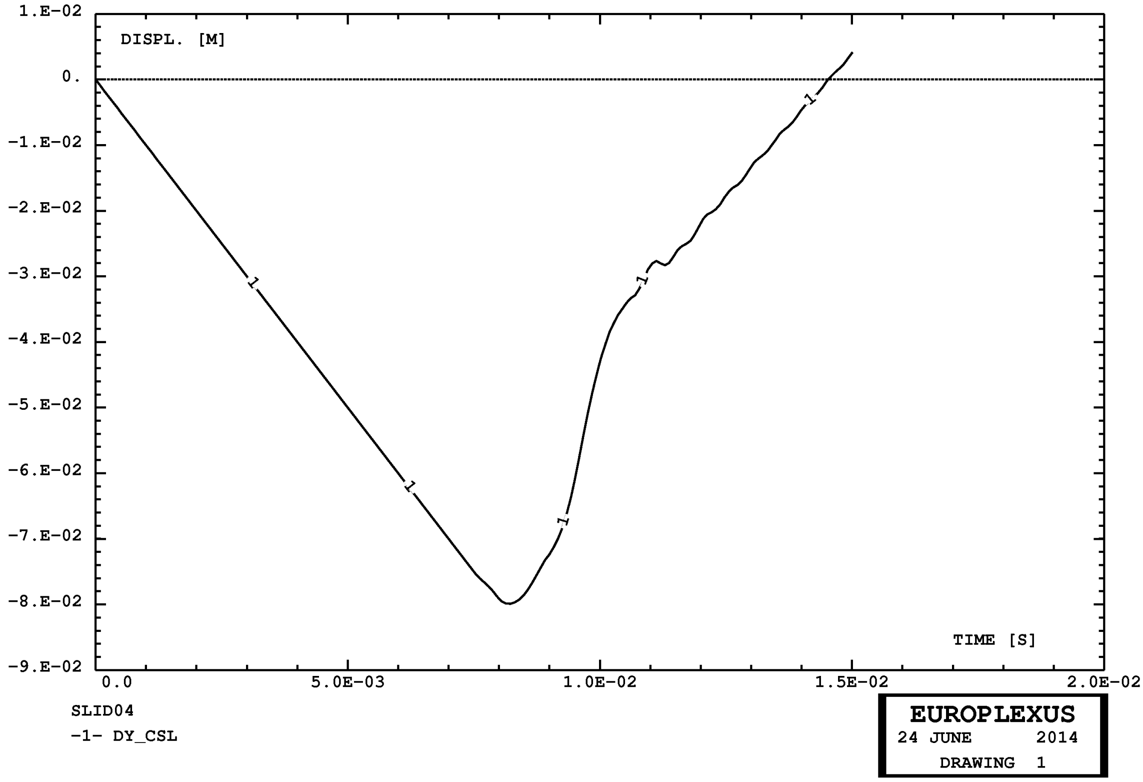


Figure 101 - Displacement of the upper block center in case SLID04.

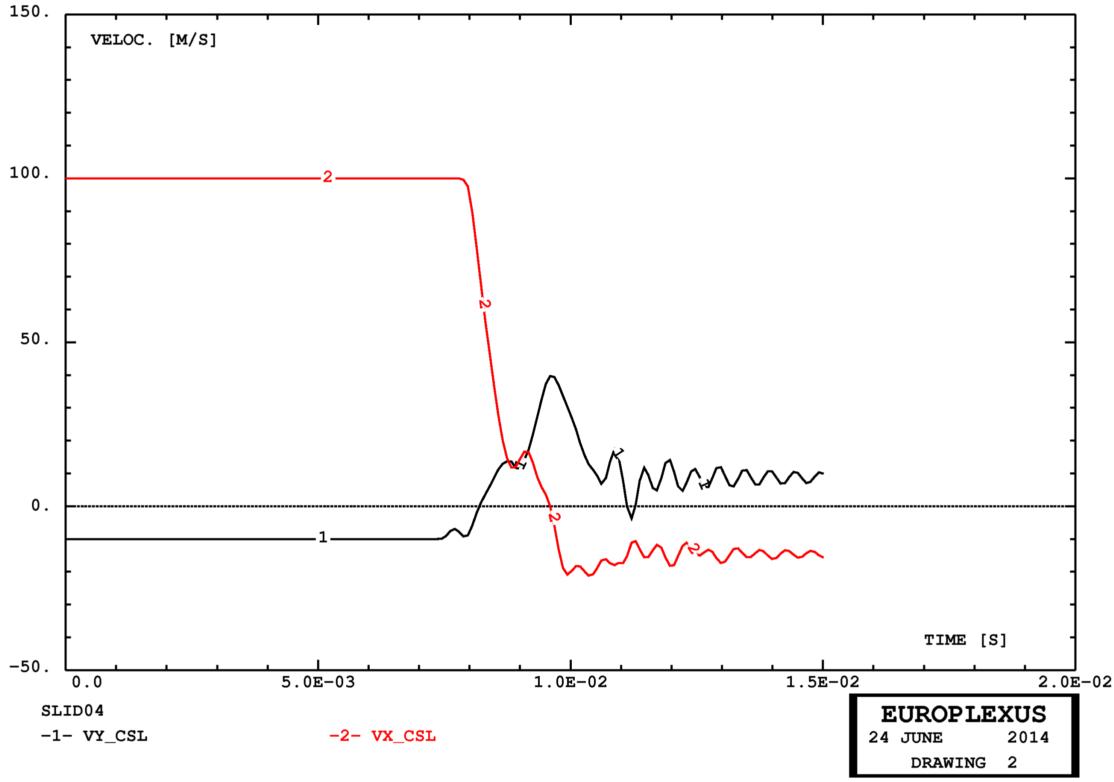


Figure 102 - Velocity components of the upper block center in case SLID04.

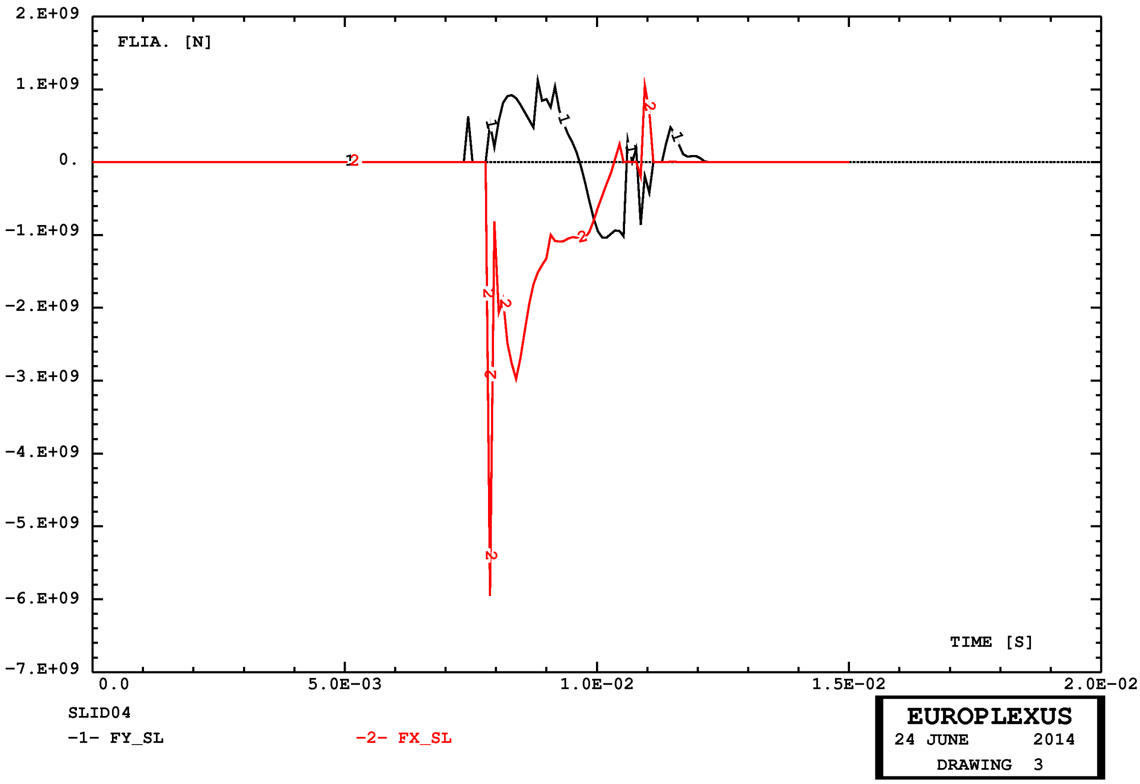


Figure 103 - Contact force components of the upper block center in case SLID04.

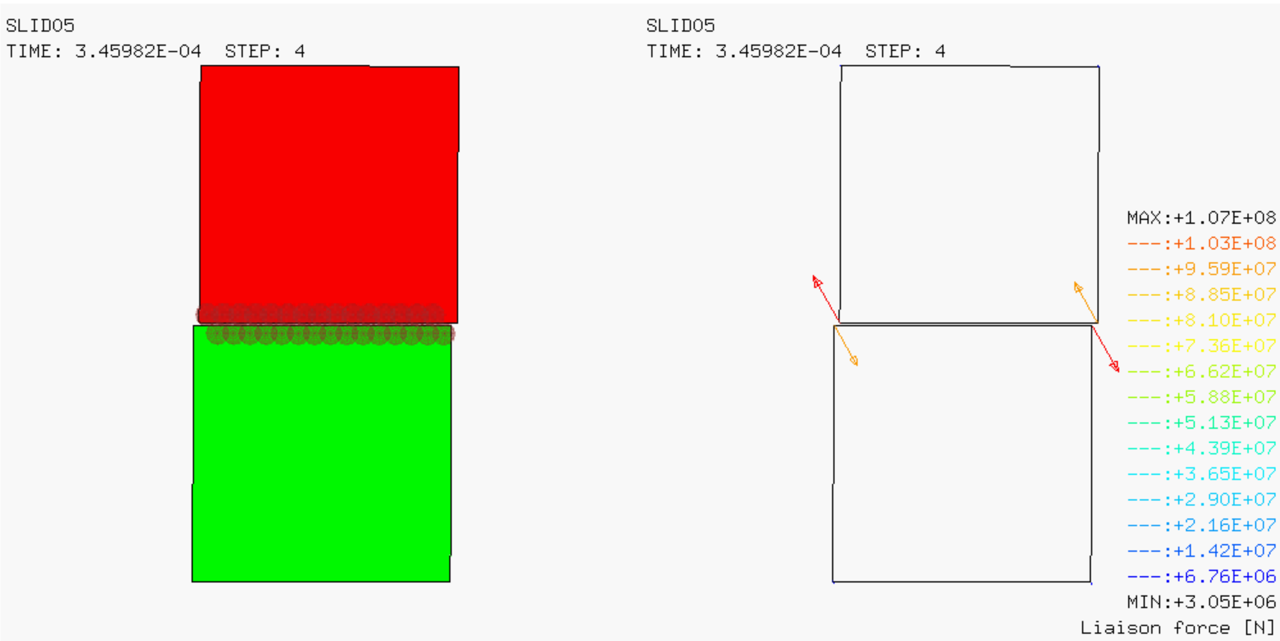
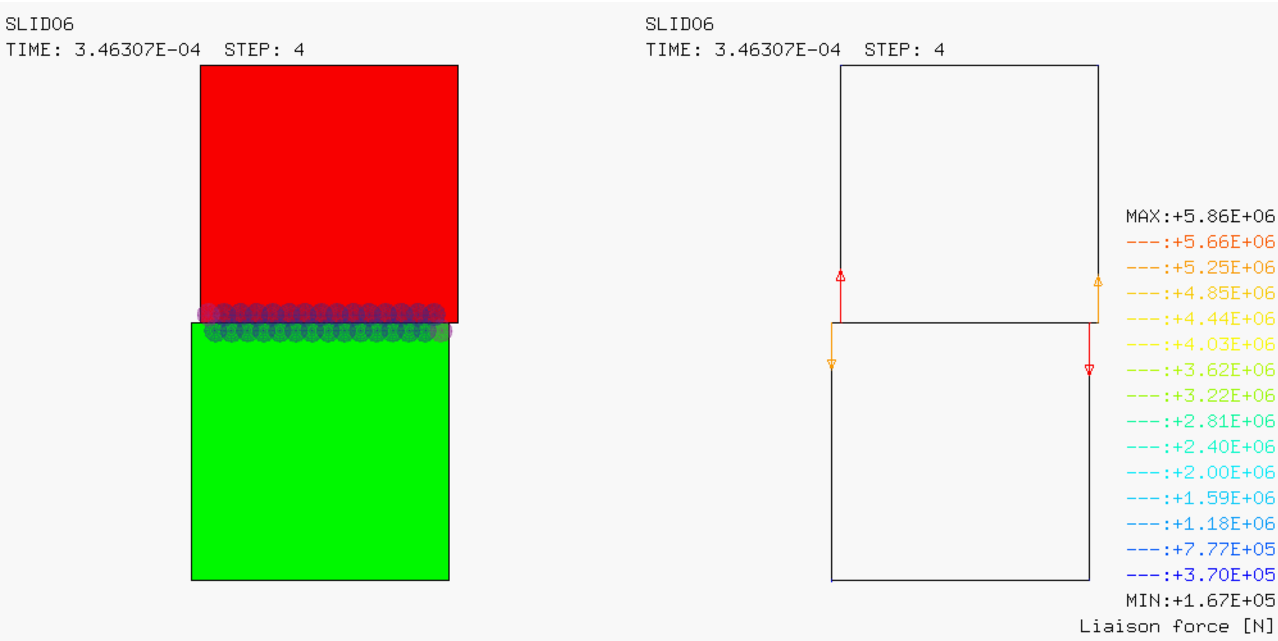
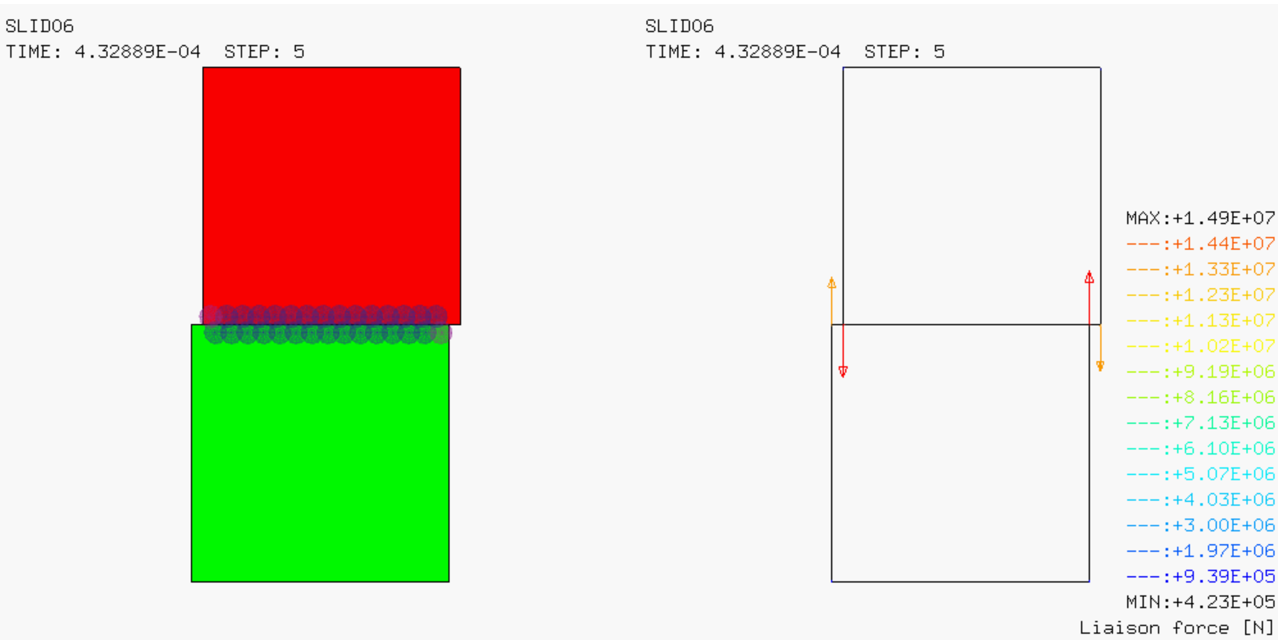


Figure 104 - Sliding contact test SLID05.



Step 4 (contact forces are repulsive)



Step 5 (contact forces become attractive)

Figure 105 - Sliding contact test SLID06.

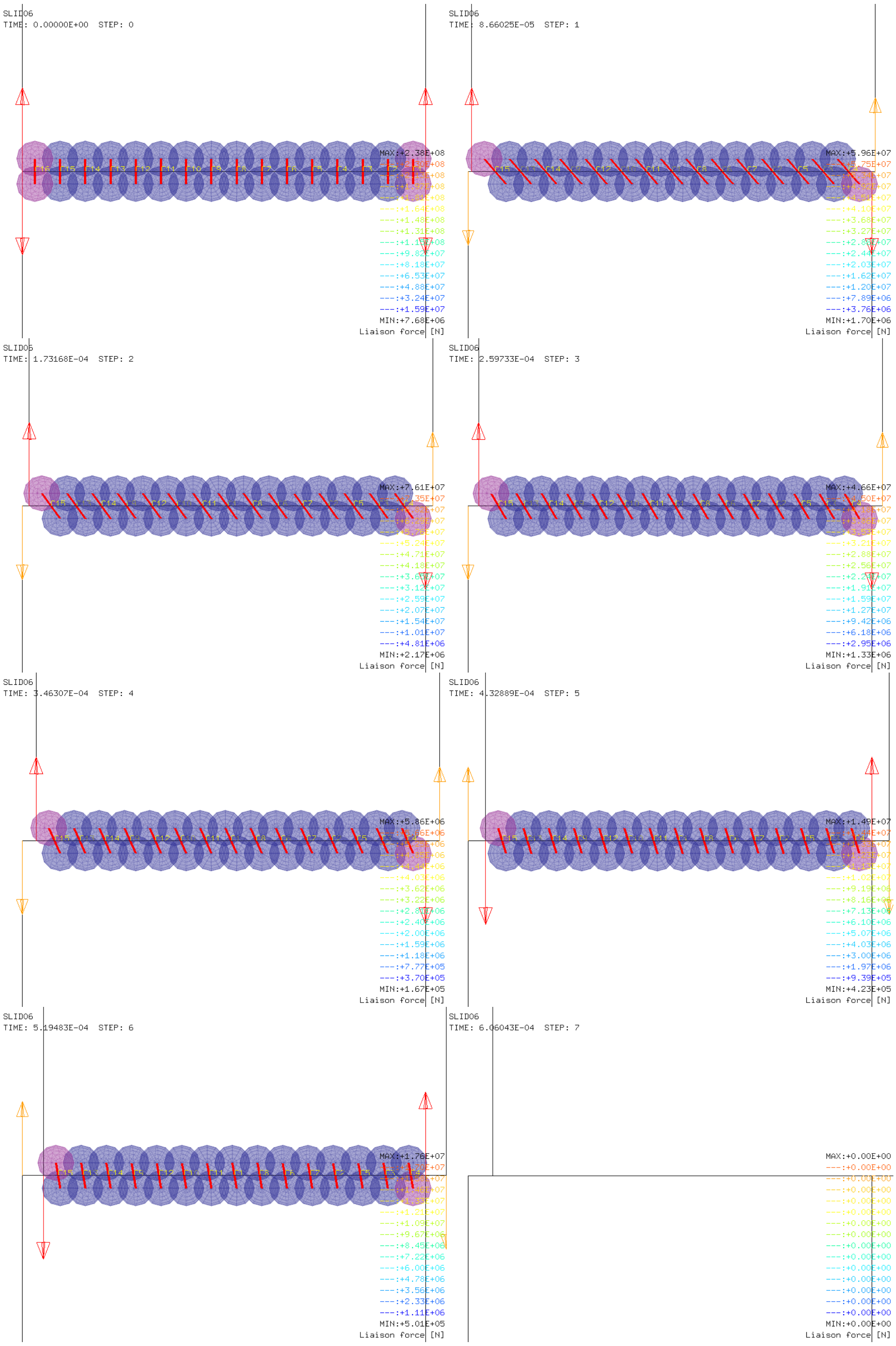
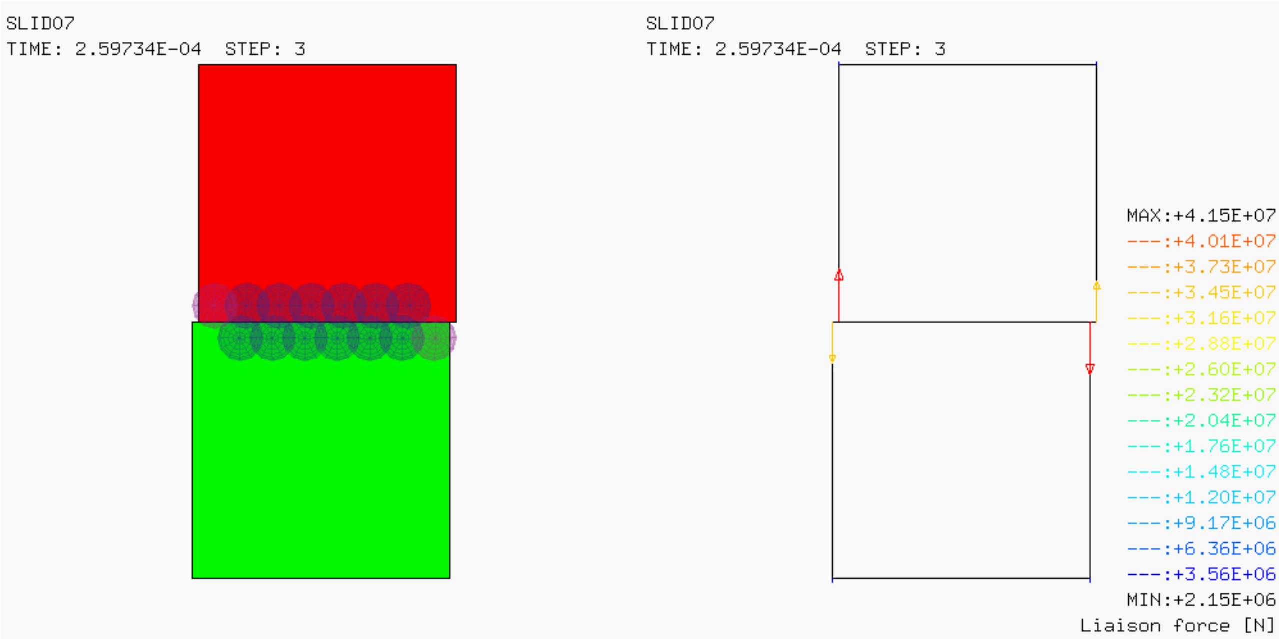
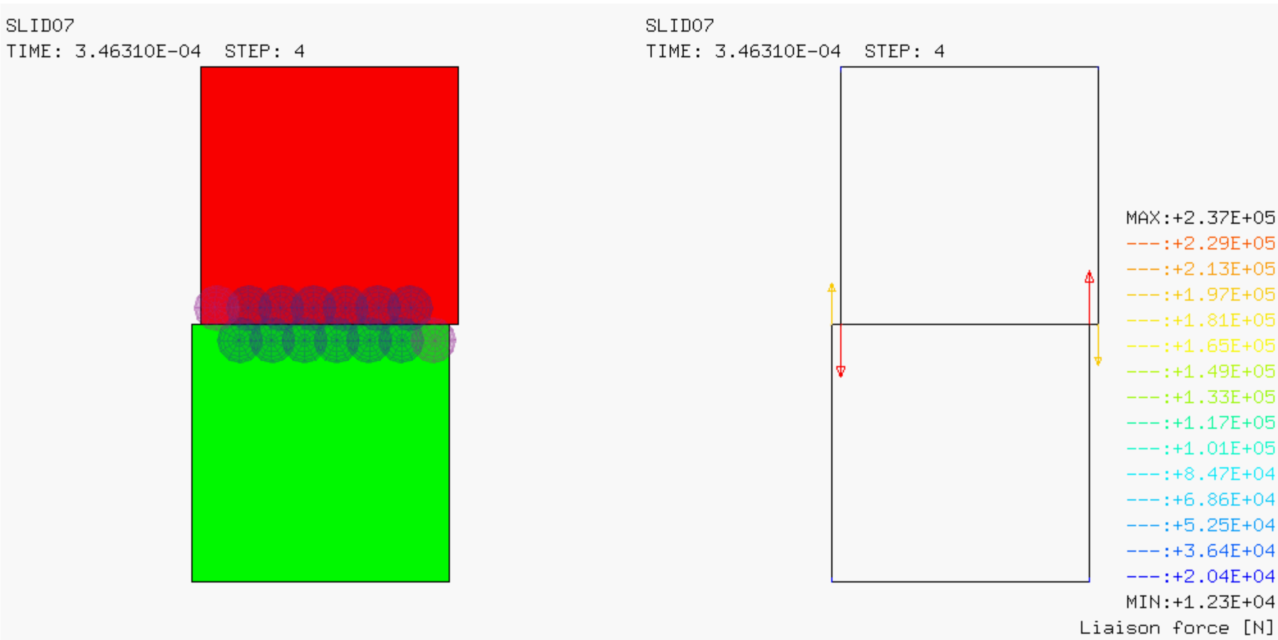


Figure 106 - Contact configurations in test SLID06 (steps 0 to 7).

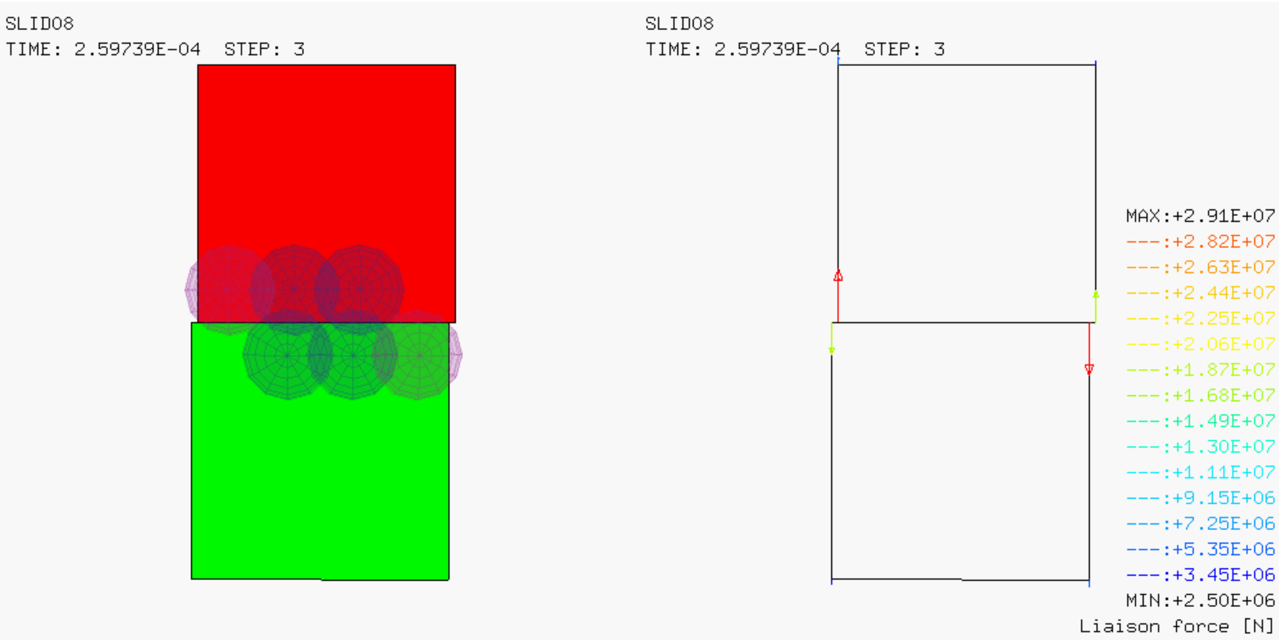


Step 3 (contact forces are repulsive)

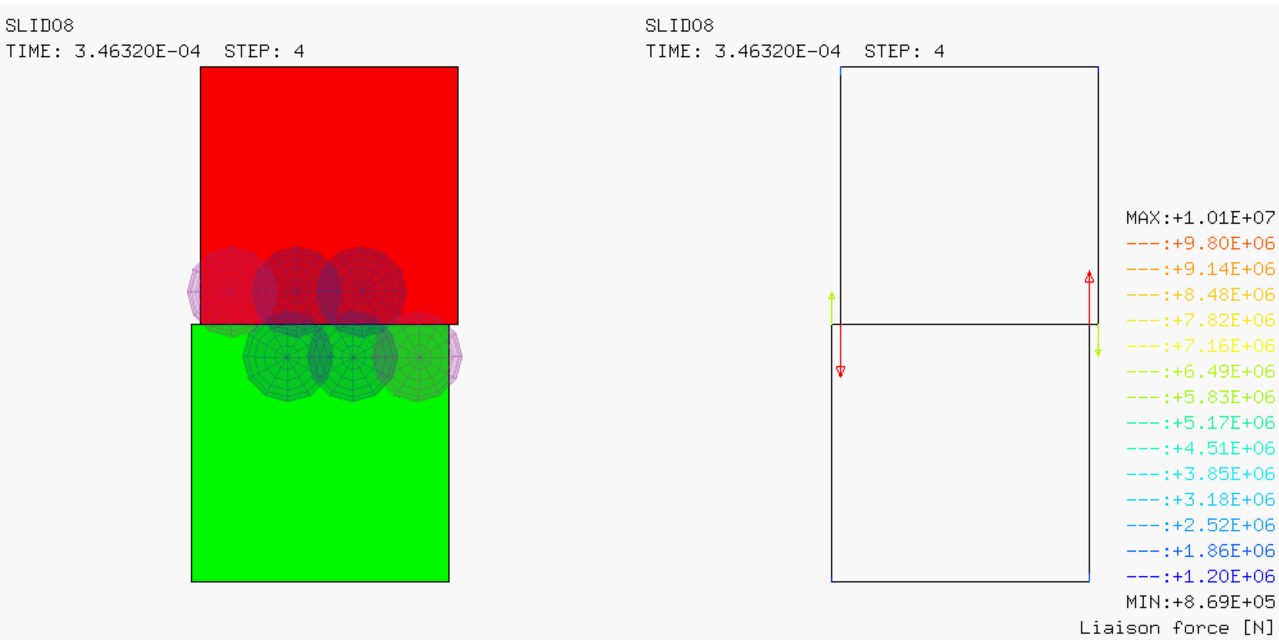


Step 4 (contact forces become attractive)

Figure 107 - Sliding contact test SLID07.

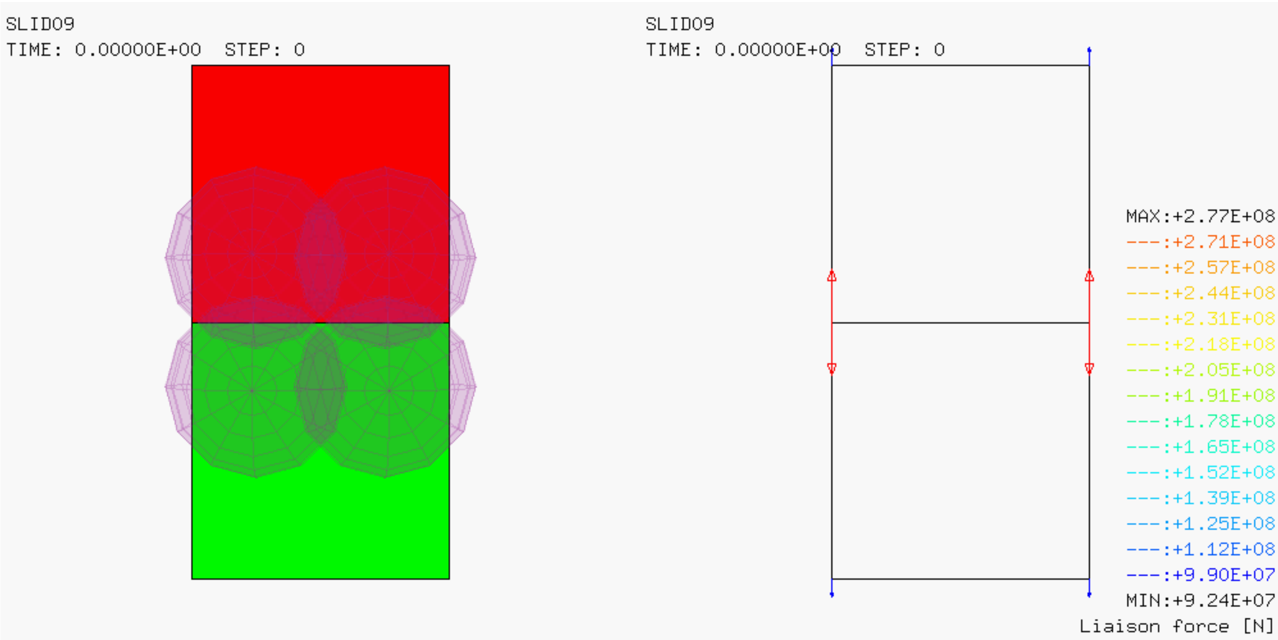


Step 3 (contact forces are repulsive)

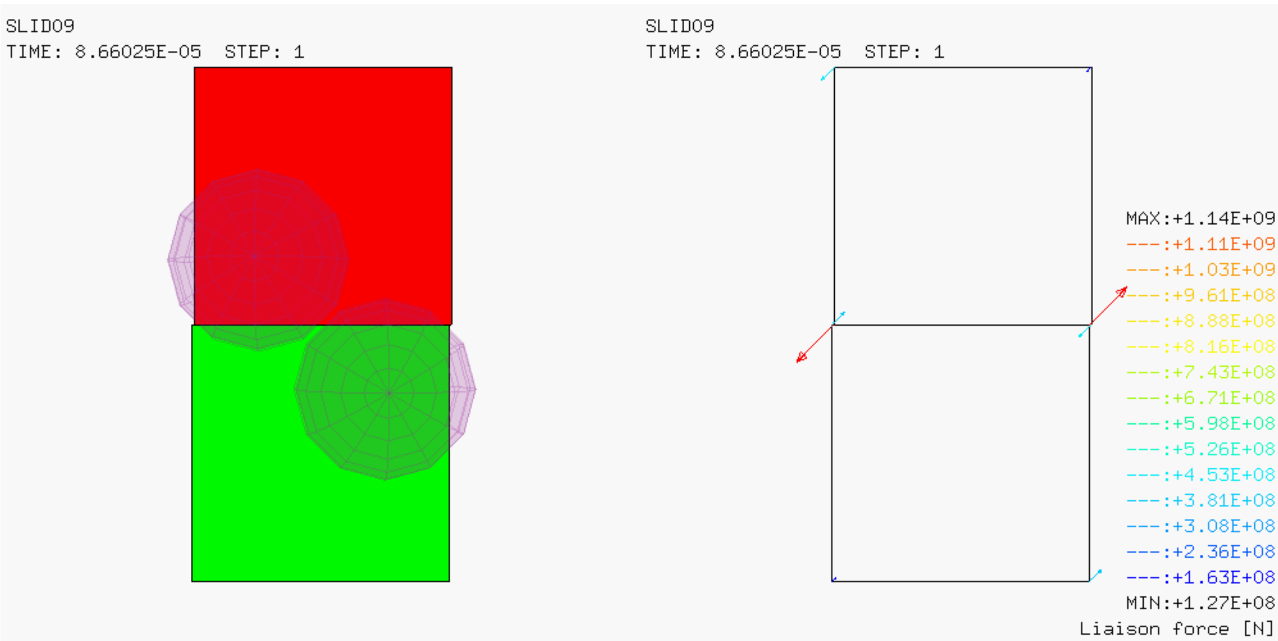


Step 4 (contact forces become attractive)

Figure 108 - Sliding contact test SLID08.



Step 0 (contact forces are repulsive and vertical)



Step 1 (contact forces become weird)

Figure 109 - Sliding contact test SLID09.

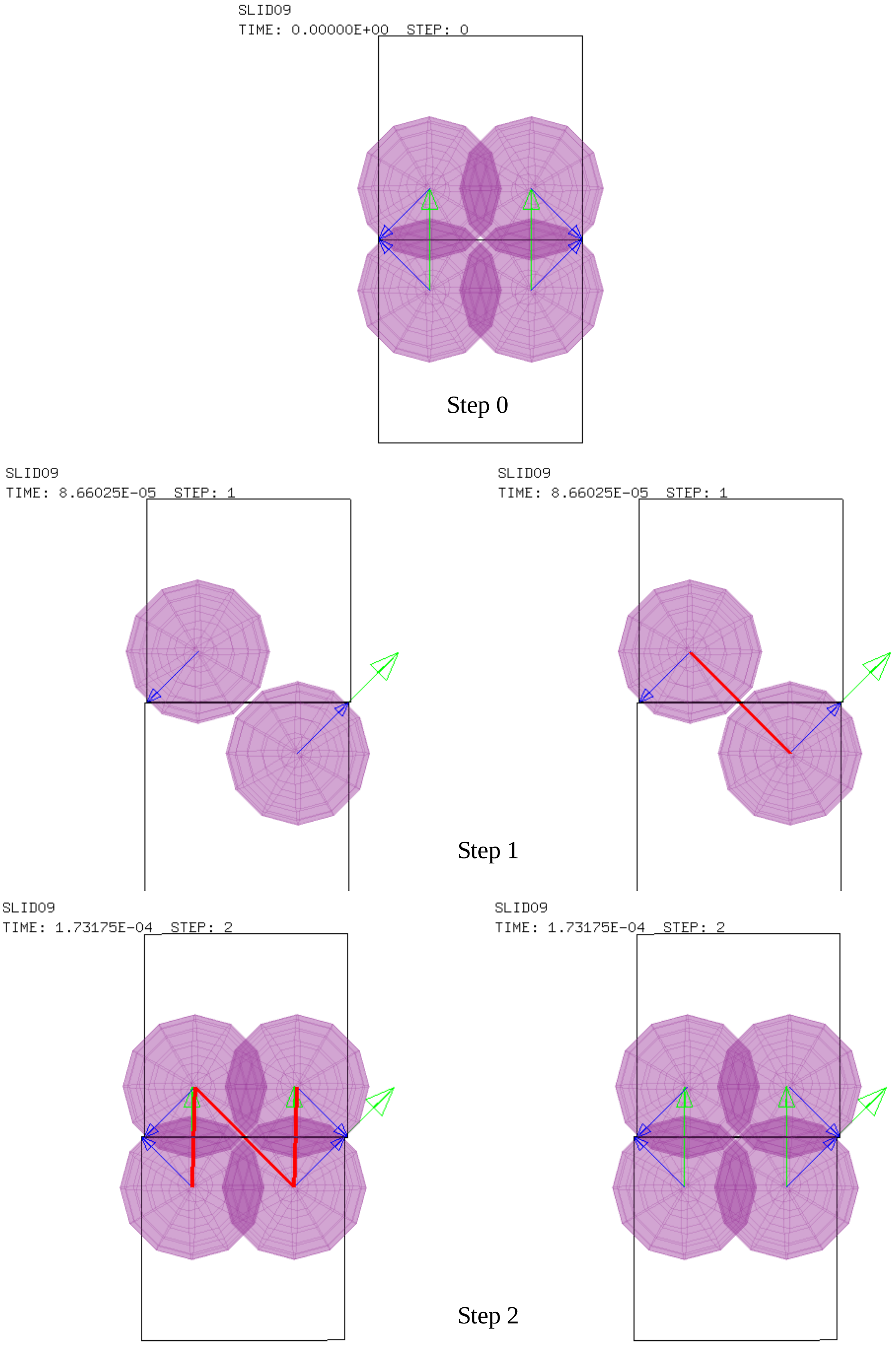
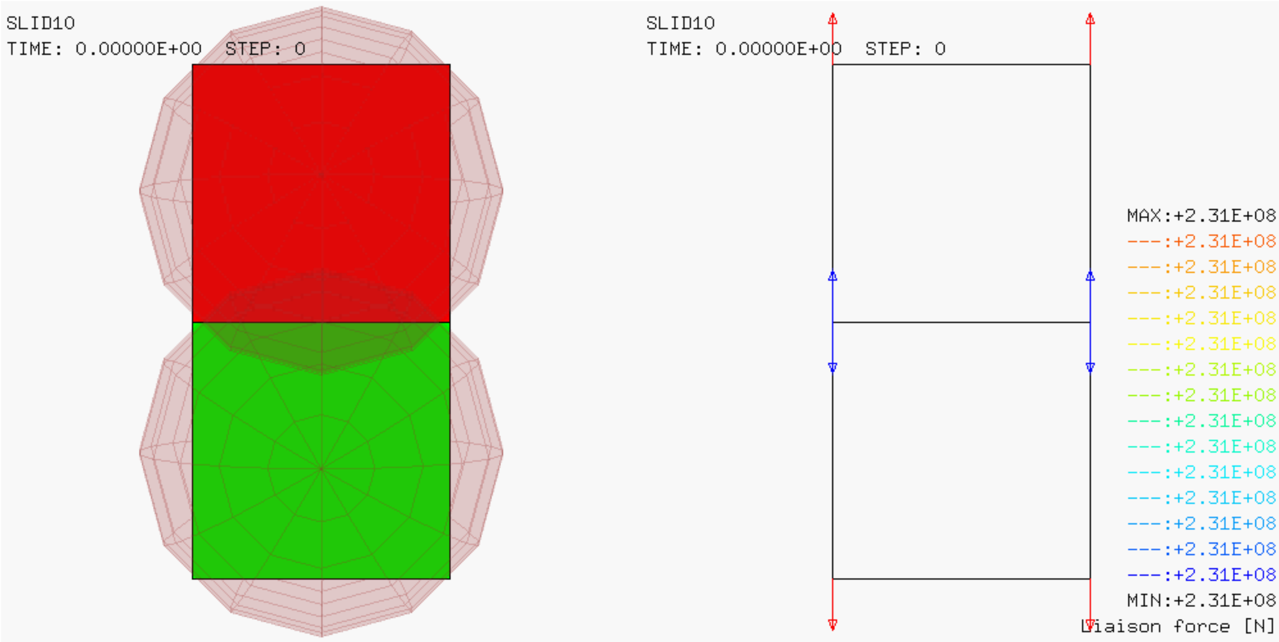
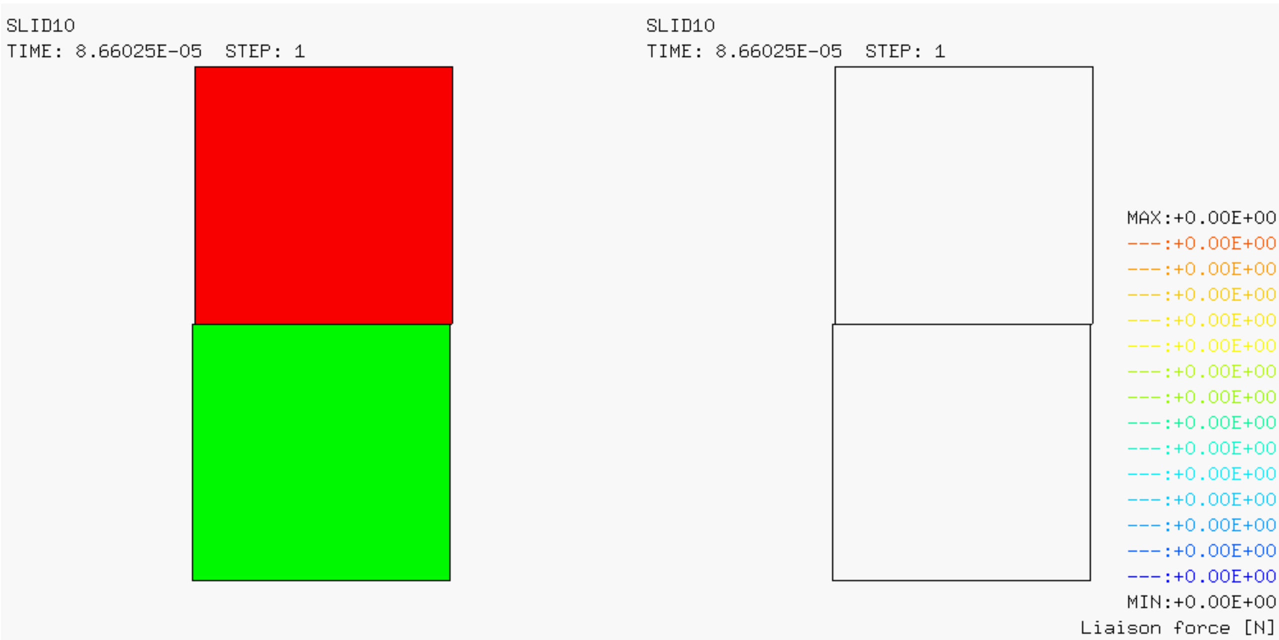


Figure 110 - Sliding contact test SLID09, details on contact conditions



Step 0 (contact forces are repulsive and vertical)



Step 1 (contact forces become weird)

Figure 111 - Sliding contact test SLID10.

5.4 Tube array crash test

The next set of tests shows a realistic contact problem, the crushing of an array of tubes (courtesy of Onera Lille), already considered in reference [15]. The initial configuration of the tubes array is shown in Figure 112.

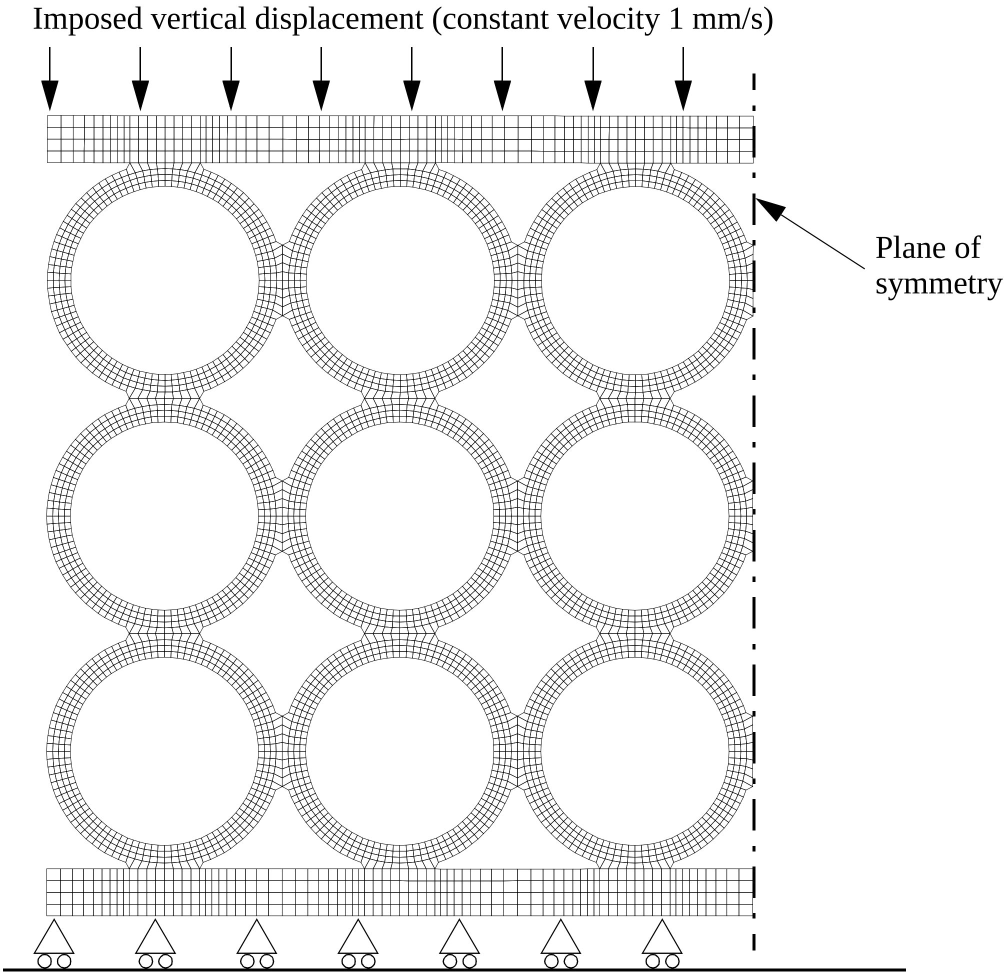


Figure 112 - Sketch of the structure crash problem (simplified version).

In reference [15] the problem has been studied by a variety of approaches, including either linear-displacement or parabolic-displacement elements and either Lagrangian or ALE formulations (the latter justified by the very large local deformations of the structure). In all cases, pinballs with a Lagrange Multipliers method (`LINK COUP`) were employed for the contact.

Some of these simulations are repeated by adding the new ASN technique, and also a penalty-based solution (rather than LM) is attempted. We choose as a reference the test CARA07 of reference [15], which is an ALE simulation using the Q42 element, although this solution was an over-stiff one (the Q42 element is a fully integrated linear-displacement quadrilateral).

The new simulations are listed in Table 10 and are described hereafter.

Name	Mesh	Notes	Steps	CPU [s]	Els*step
CARN07	4080 Q42	LINK DECO PINB PENA SELF DMIN 0.1 PINS GRID EQVF ASN NORB	89,289 1 s	1,681	3.64×10^{8}
CARO07	4080 Q42	LINK COUP PINB SELF DMIN 0.1 PINS GRID EQVF ASN NORB	89,950 1 s	1,543	3.67×10^{8}

Table 10 - Tube array crash tests.

CARN07

This test is a repetition of CARA07 of reference [15], but uses de-coupled boundary conditions instead of coupled (LM) ones. This means that the blockages and symmetries are imposed via `LINK DECO BLOQ` (instead of LINK `COUP BLOQ`), and that the contact is treated by `LINK DECO PINB PENA` instead of `LINK COUP PINB`. A penalty coefficient `SFAC 1.0` is chosen. In addition, the ASN algorithm is activated by adding the option `OPTI PINS ASN NORB`. The latter keyword disables the treatment of a-priori rebound, which would be active by default, since normally with the penalty method the contact force should be applied also during the rebound phase, as long as there is penetration.

Figure 113 shows the initial configuration with nodal ASNs (in magenta) and parent pinball ASNs (in blue). Figure 114 shows the nodal ASNs in an intermediate configuration, at time 0.615 s. Figures 115 and 116 show the contact normal directions and the contact forces, respectively, for some contacting descendent pinballs near one of the folds at 0.76 s. Note that the contact forces are directed along the normal, and the normal is very reasonably perpendicular to the local contact surface (which is not always the case with standard pinballs without the ASN algorithm).

Figures 117 and 118 show the contacts and the equivalent plastic strains in the final configuration, i.e. at time 1.0 s.

Figure 119 shows the upper and lower plate crushing forces. As already noted in the simulations of reference [15] the two curves are practically identical, except for a very short initial period where some slight dynamic effects are present. Figure 120 compares the crushing forces in solutions CARA07 (no ASN, coupled links and in particular LM formulation for the contacts), shown in green, and CARN07 (ASN, de-coupled links and in particular penalty formulation for the contacts), shown in black. The penalty/ASN solution shows a reduction of the crushing force in the final part of the transient. The final value of the force passes from 6,100 to 5,400, a reduction of 11%, and should therefore be in better agreement with the experimental result.

CARO07

This test is a repetition of CARN07 but uses the coupled (LM) formulation for the links (`LINK COUP BLOQ` and `LINK COUP PINB`). The difference with respect to case CARA07 is therefore only in the ASN algorithm for the determination of the contact normal. The scope is to see whether the differences observed in the previous solution with respect to CARA07 are due to the penalty formulation, or to the ASN (or to both).

Figures 121 and 122 show the contacts and the equivalent plastic strains in the final configuration, i.e. at time 1.0 s.

Figure 123 compares the crushing forces in solutions CARA07 (in green), CARN07 (in red) and CARO07 (in black). The latter solution shows only a very slight reduction of final crushing force with respect to case CARA07. therefore, most of the reduction in force observed in the previous solution comes from the penalty formulation (vs. the LM formulation) and the ASN seems to have only a minor effect in this particular example.

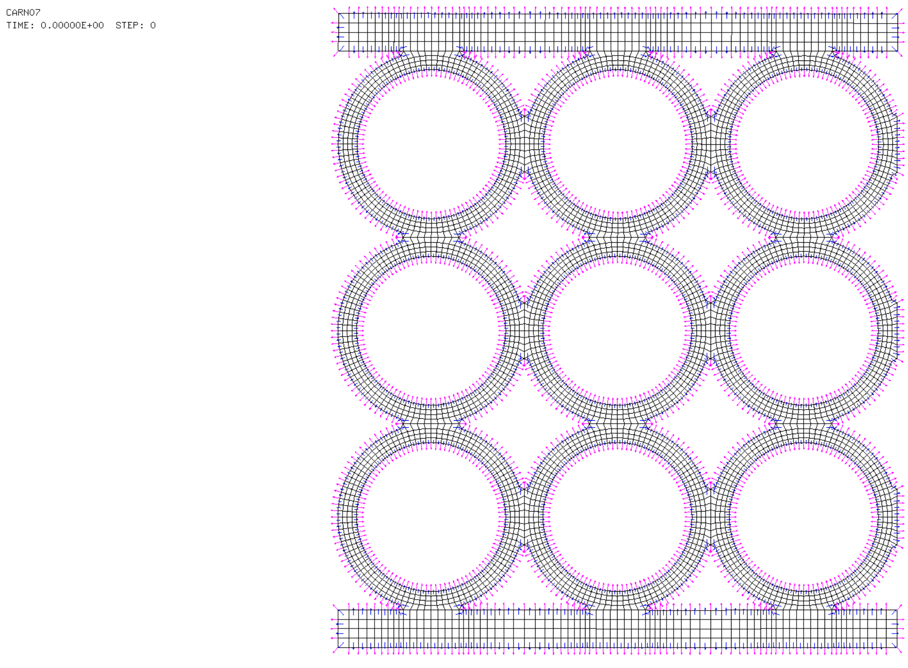


Figure 113 - Nodal ASNs and parent pinball ASNs in case CARN07 (initial configuration).

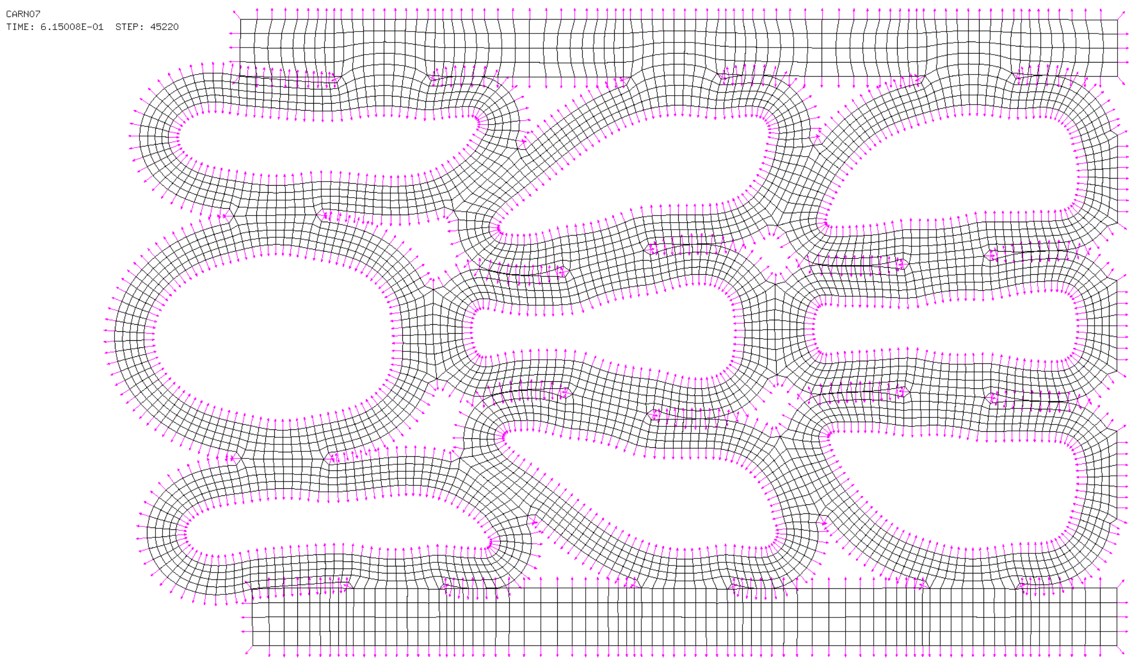


Figure 114 - Nodal ASNs in case CARN07 at 0.615 s.

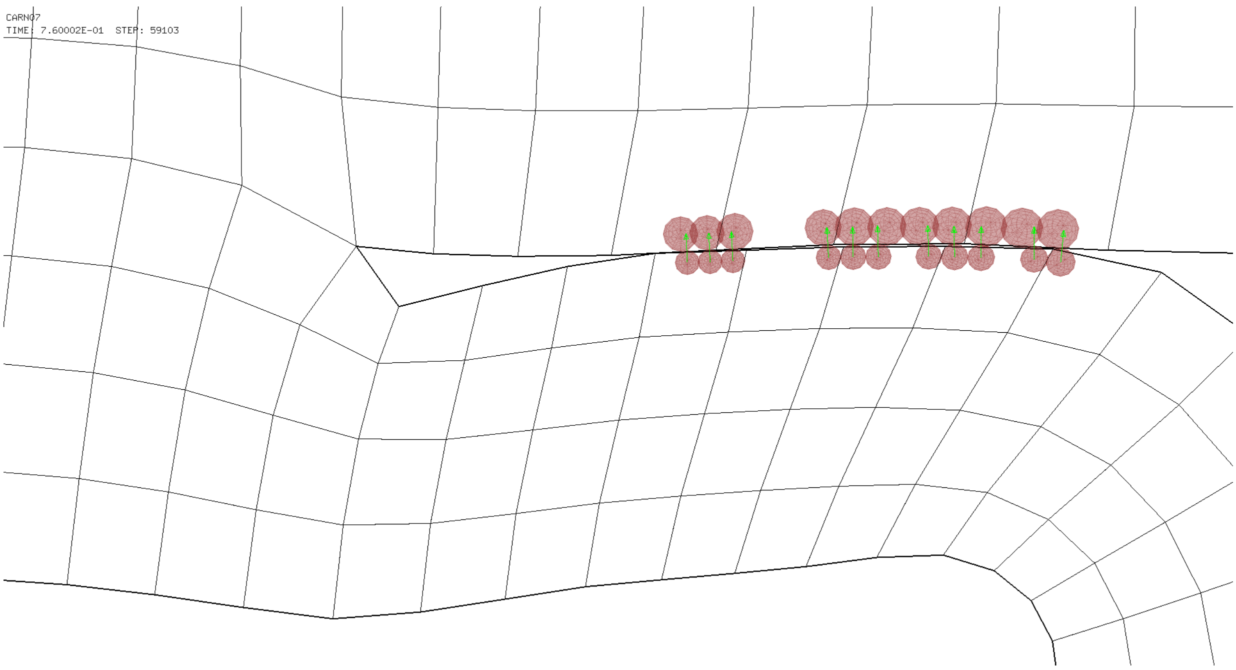


Figure 115 - Contact normals in case CARN07 at 0.76 s.

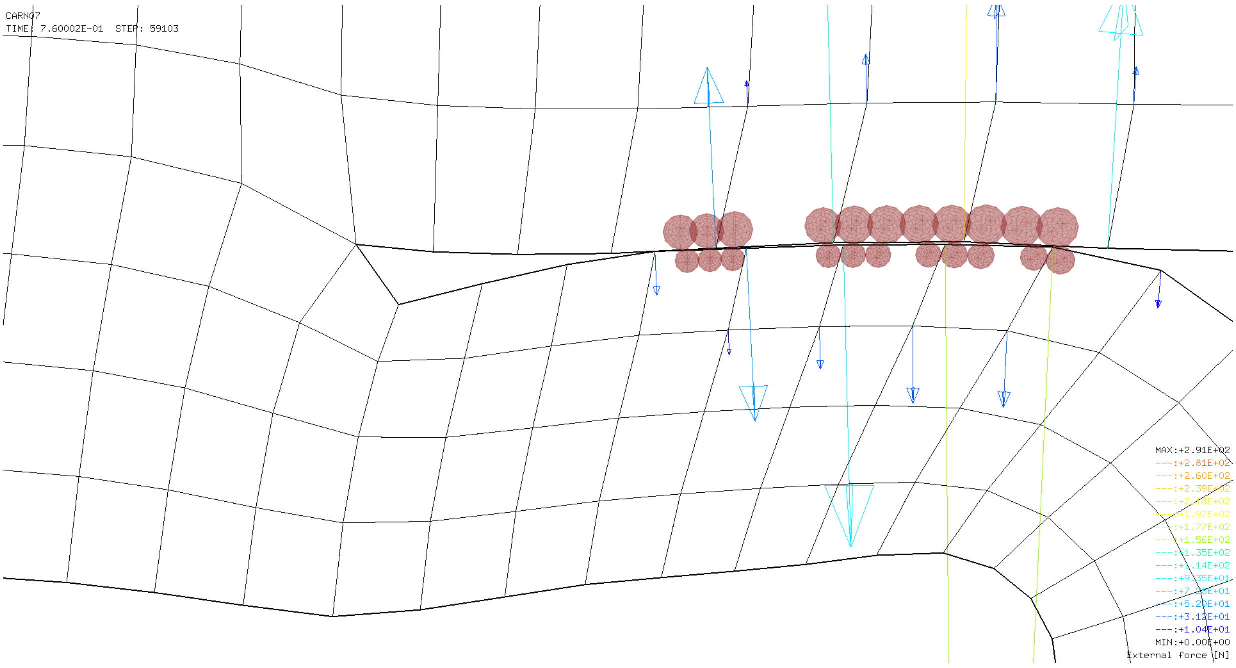


Figure 116 - Contact forces in case CARN07 at 0.76 s.

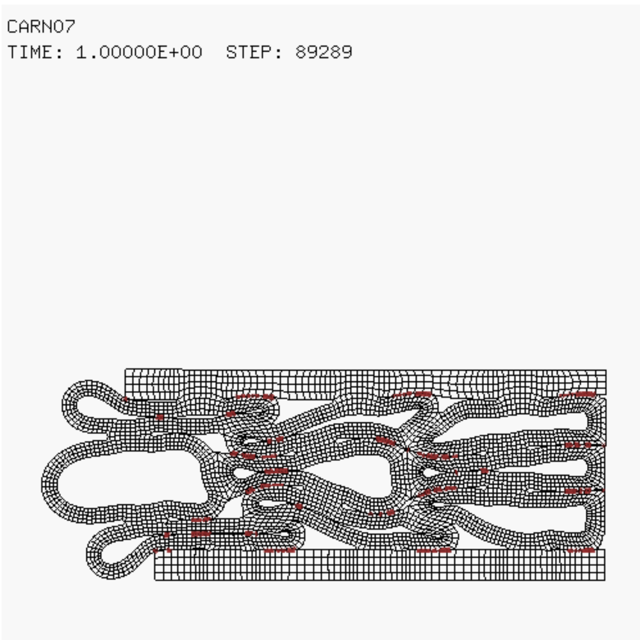


Figure 117 - Final contacts in case CARN07.

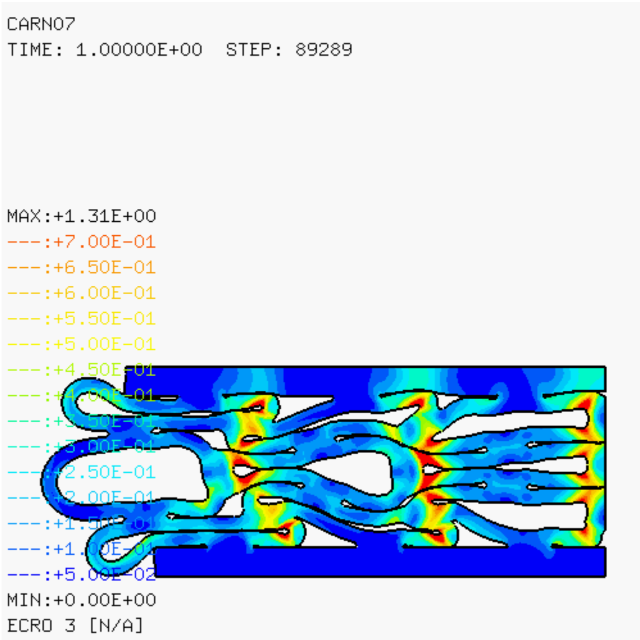


Figure 118 - Final plastic strains in case CARN07.

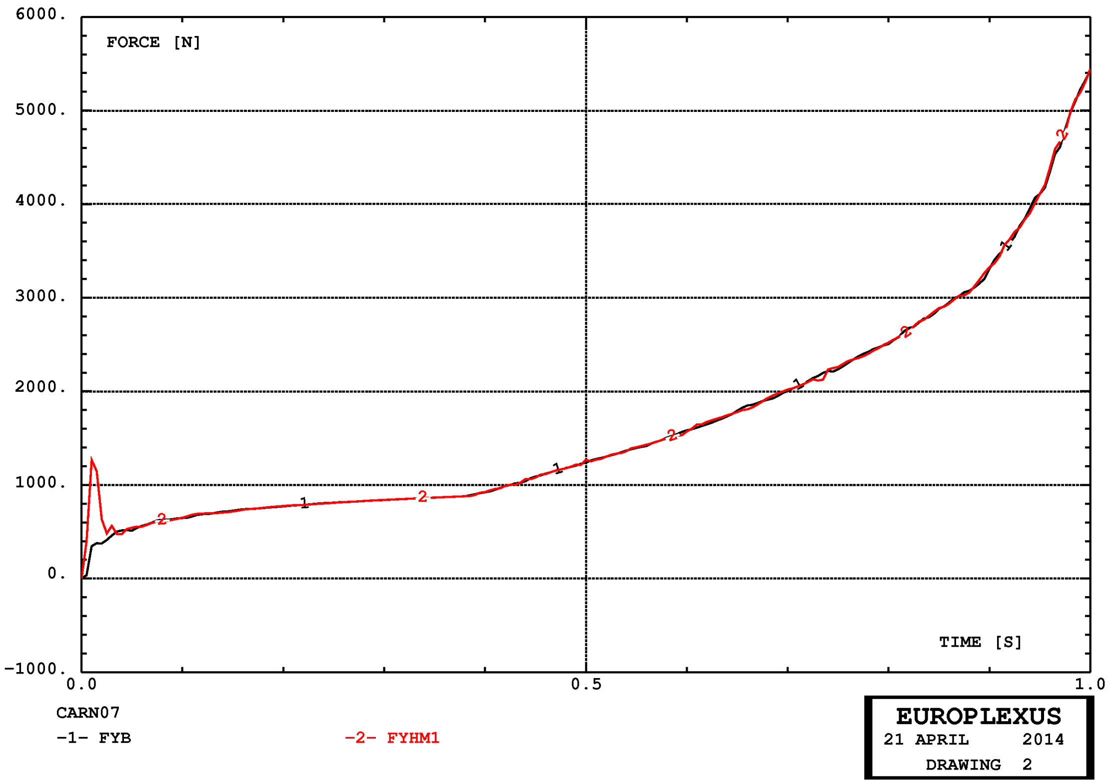


Figure 119 - Upper and lower crushing forces in case CARN07.

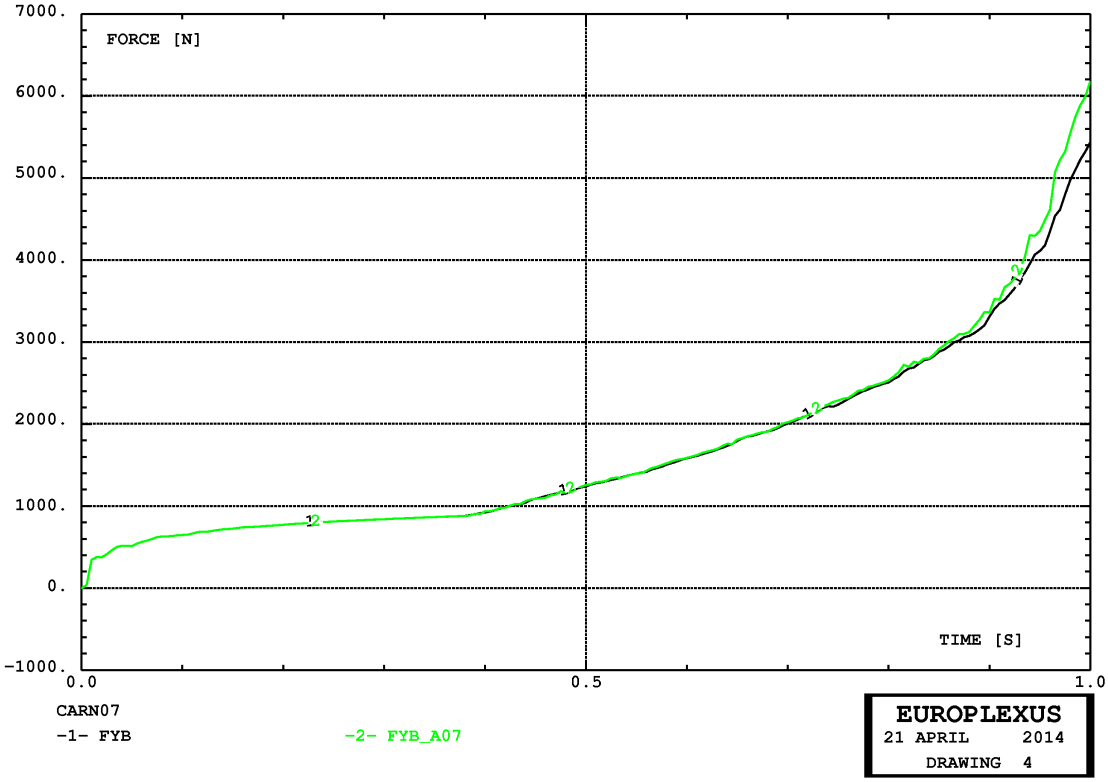


Figure 120 - Crushing forces in cases CARA07 and CARN07.

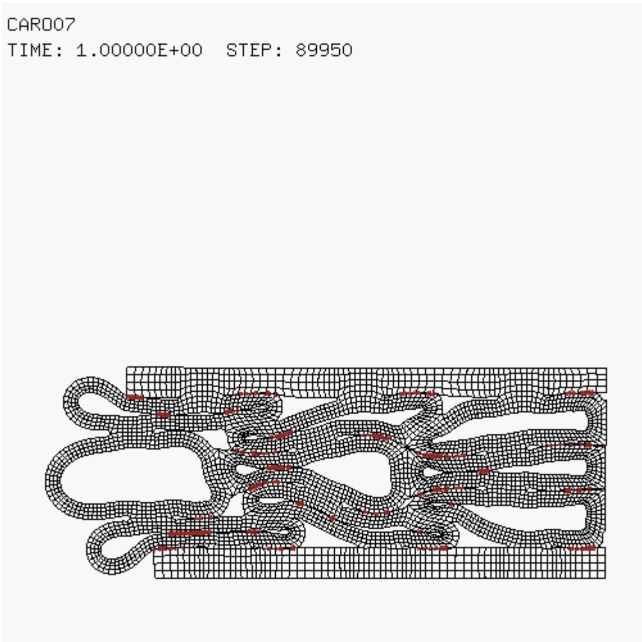


Figure 121 - Final contacts in case CARO07.

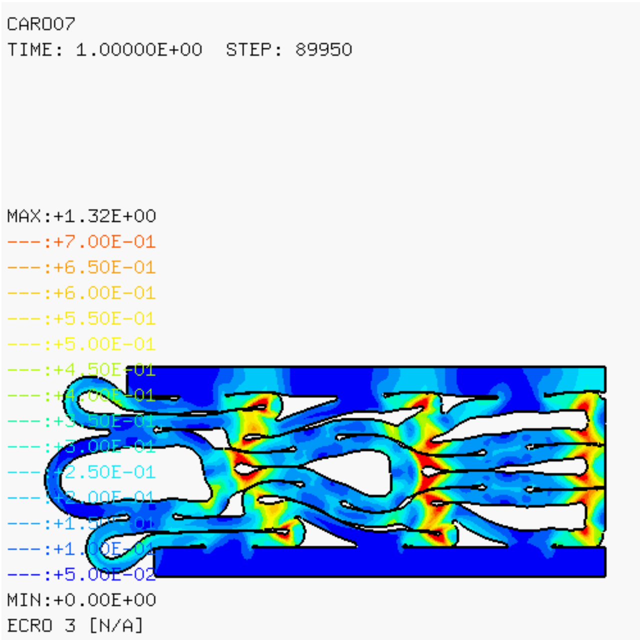


Figure 122 - Final plastic strains in case CARO07.

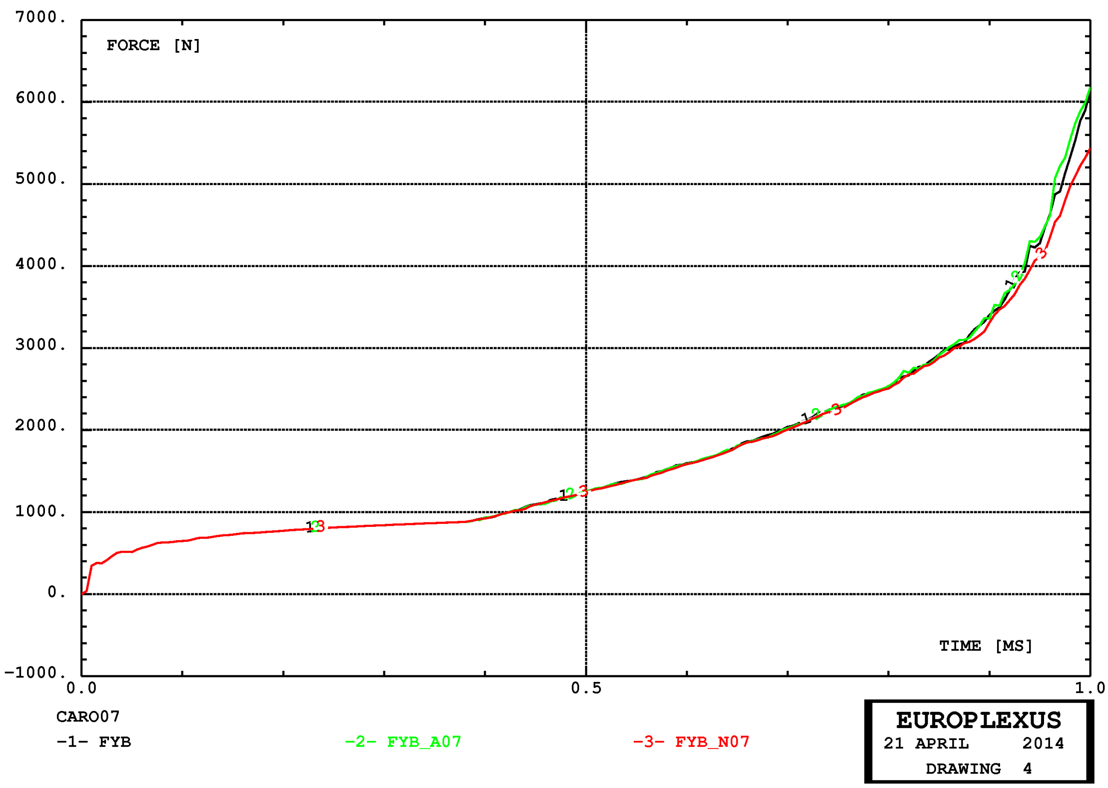


Figure 123 - Crushing forces in cases CARA07, CARN07 and CARO07.

6. References

[1] Hallquist JO, Goudreau GL, Benson DJ. Sliding interfaces with contact-impact in large-scale Lagrangian computation. Computer Methods in Applied Mechanics and Engineering 1985; **51** : 107-137.

[2] Benson DJ, Hallquist JO. A single surface contact algorithm for the postbuckling analysis of shell structures. Report to the University of California at San Diego, CA, 1987.

[3] Belytschko T, Law SE. An Assembled Surface Normal Algorithm for Interior Node Removal in Three-Dimensional Finite Element Meshes. Engineering with Computers 1985 ; **1** : 55-60.

[4] Belytschko T, Lin JI. A Three-Dimensional Impact-Penetration Algorithm with Erosion. Computers and Structures 1987 ; **25** : 95-104.

[5] Belytschko T, Neal MO. Contact-Impact by the Pinball Algorithm with Penalty, Projection and Lagrangian Methods, in Computational Techniques for Contact, Impact, Penetration and Perforation of Solids, Winter Annual Meeting of the American Society of Mechanical Engineers, San Francisco, California, December 10-15, 1989 ; AMD **103** : 97-140.

[6] Belytschko T, Neal MO. The Vectorized Pinball Contact Impact Routine. SMiRT-10 Conference, 1989.

[7] Sarwas RE. Hidden Line Elimination and a Modified Pinball Algorithm for Finite Element Contact-Impact Problems with Shell Elements. Master's Thesis, Northwestern University, Evanston, Illinois, December 1989.

[8] Belytschko T, Neal MO. Contact-Impact by the Pinball Algorithm with Penalty and Lagrangian Methods. International Journal for Numerical Methods in Engineering 1991; 31 : 547-572.

[9] Belytschko T, Yeh IS. The Splitting Pinball Method for Contact-Impact Problems. Computer Methods in Applied Mechanics and Engineering 1993; 105 : 375-393.

[10] Plaskacz E, Belytschko T, Chiang HY. Contact-Impact Simulations on Massively Parallel SIMD Supercomputers. Argonne National Laboratory Report ANL/CP-76686, 1993.

[11] Lepareux M, Schwab B, Hoffmann A, Jamet P, Bung H. Un programme général pour l'analyse dynamique rapide-Cas des tuyauteries. Colloque: Tendances Actuelles en Calcul des Structures, Bastia, 6-8 November, 1985.

[12] Casadei F. A Hierarchic Pinball Method for Contact-Impact in Fast Transient Dynamics. VI Congresso Nazionale della Società Italiana di Matematica Applicata e Industriale (SIMAI 2002), Chia (Cagliari), Italy, 27-31 May 2002.

[13] Casadei F. A General Impact-Contact Algorithm Based on Hierarchic Pinballs for the EURO-

PLEXUS Software System. Joint Research Center Report N. I.03.176, December 2003.

[14] Casadei F. Validation of the EUROPLEXUS Pinball Impact-Contact Model on an Indentation Problem. Technical Note N. I.05.51, July 2005.

[15] Casadei F, Larcher M, Valsamos G, Langrand B. Pinball-based Contact-Impact Model with Parabolic Elements in EUROPLEXUS. Technical Note JRC Pubsy N. JRC89913, EUR Report 26629 EN, 2014.

[16] "*EUROPLEXUS User's Manual*", on-line version: http://europlexus.jrc.ec.europa.eu.

[17] Casadei F, Faucher V. Contact-impact in explicit fast transient dynamics by the hierarchic pinball method with Lagrange multipliers. Journal paper in preparation, 2014.

7. Appendix A - Closest points on two segments

To treat contact between 3D bars, it is useful to compute the closest points on the bars. Let us first consider the problem of finding the closest points on two (infinite) lines L_1, L_2, shown in Figure 124. We assume for simplicity that the two lines are not parallel.

If P_0, Q_0 are points on the lines and $\underline{u}$, $\underline{v}$ are vectors directed along the lines, and s, t are (scalar) parameters, then the two lines can be represented by the following parametric vector equations:

$$\begin{aligned} L_1: &\qquad P(s) = P_0 + s\underline{u} \\ L_2: &\qquad Q(t) = Q_0 + t\underline{v} \end{aligned} . \tag{92}$$

Let $\underline{w}(s, t) = P(s) - Q(t)$ be a generic vector between points on the two lines. If we denote $P(s_c)$ and $Q(t_c)$ the points on the two lines which are <u>closest</u> to each other, then the vector $\underline{w}_c(s_c, t_c)$ has the minimum length for all s and t, and <u>is perpendicular to both lines</u>. This vector is unique, i.e. no other vector $\underline{w}(s, t)$ has this property. This condition is expressed by the system of equations:

$$\begin{cases} \underline{u} \cdot \underline{w}_c = 0 \\ \underline{v} \cdot \underline{w}_c = 0 \end{cases} . \tag{93}$$

By replacing the expression of $\underline{w}_c$:

$$\underline{w}_c = \underline{w}(s_c, t_c) = P(s_c) - Q(t_c) = P_0 + s_c\underline{u} - Q_0 - t_c\underline{v} = \underline{w}_0 + s_c\underline{u} - t_c\underline{v} \tag{94}$$

(where $\underline{w}_0 = P_0 - Q_0$) into (93) one gets:

$$\begin{cases} (\underline{u} \cdot \underline{u})s_c - (\underline{u} \cdot \underline{v})t_c = -\underline{u} \cdot \underline{w}_0 \\ (\underline{v} \cdot \underline{u})s_c - (\underline{v} \cdot \underline{v})t_c = -\underline{v} \cdot \underline{w}_0 \end{cases} . \tag{95}$$

Then by letting the known terms:

$$\begin{aligned} a &= \underline{u} \cdot \underline{u} \\ b &= \underline{u} \cdot \underline{v} = \underline{v} \cdot \underline{u} \\ c &= \underline{v} \cdot \underline{v} \\ d &= \underline{u} \cdot \underline{w}_0 \\ e &= \underline{v} \cdot \underline{w}_0 \end{aligned} \tag{96}$$

we obtain the two linear algebraic equations in the two unknown scalars s_c, t_c:

$$\begin{cases} as_c - bt_c = -d \\ bs_c - ct_c = -e \end{cases} . \tag{97}$$

By solving we obtain:

$$\begin{aligned}
s_c &= \frac{bt_c - d}{a} \\
\frac{b^2 t_c - bd}{a} - ct_c &= -e \\
b^2 t_c - bd - act_c &= -ae \\
act_c - b^2 t_c + bd &= ae \\
(ac - b^2)t_c &= ae - bd \\
t_c &= \frac{ae - bd}{ac - b^2}
\end{aligned} \quad . \tag{98}$$

$$s_c = \frac{b}{a}\left(\frac{ae - bd}{ac - b^2}\right) - \frac{d}{a} = \frac{b(ae - bd) - d(ac - b^2)}{a(ac - b^2)} = \frac{abe - b^2 d - acd + b^2 d}{a(ac - b^2)} = \frac{be - cd}{ac - b^2}$$

Summarizing, it is:

$$\begin{aligned}
s_c &= \frac{be - cd}{ac - b^2} \\
t_c &= \frac{ae - bd}{ac - b^2}
\end{aligned} \tag{99}$$

whenever the denominator $ac - b^2 \neq 0$. One may note that, from (96):

$$\begin{aligned}
ac - b^2 &= (\underline{u} \cdot \underline{u})(\underline{v} \cdot \underline{v}) - (\underline{u} \cdot \underline{v})^2 = \\
&= \|\underline{u}\|^2 \|\underline{v}\|^2 - (\|\underline{u}\| \|\underline{v}\| \cos\theta)^2 = \\
&= \|\underline{u}\|^2 \|\underline{v}\|^2 - \|\underline{u}\|^2 \|\underline{v}\|^2 \cos^2\theta = \\
&= \|\underline{u}\|^2 \|\underline{v}\|^2 (1 - \cos^2\theta) = \\
&= \|\underline{u}\|^2 \|\underline{v}\|^2 \sin^2\theta = \\
&= (\|\underline{u}\| \|\underline{v}\| \sin\theta)^2 \geq 0
\end{aligned} \quad , \tag{100}$$

where θ is the angle between the vectors $\underline{u}$ and $\underline{v}$, i.e. the denominator is non-negative. The denominator is zero when $\sin\theta = 0$, i.e. for $\theta = 0$. In this case, the two equations are linearly dependent, the two lines are parallel (contrary to the assumption made above), the distance between the lines is constant and there are infinite couples of points with the minimum distance. One can find the distance by fixing the value of one of the parameters and by using any of the equations to find the other one. For example, by selecting $s_c = 0$ we get from (97):

$$t_c = \frac{d}{b} = \frac{e}{c} . \tag{101}$$

Having solved for s_c and t_c we have the points $P(s_c)$ and $Q(t_c)$ where the two lines L_1 and L_2 are closest. Then the (minimum) distance between the lines is, by using (94):

$$\begin{aligned} D(L_1, L_2) &= \|\underline{w}_c\| = \\ &= \|P(s_c) - Q(t_c)\| = \\ &= \|\underline{w}_0 + s_c\underline{u} - t_c\underline{v}\| = \\ &= \left\|(P_0 - Q_0) + \frac{(be - cd)\underline{u} - (ae - bd)\underline{v}}{ac - b^2}\right\| \end{aligned} \quad . \tag{102}$$

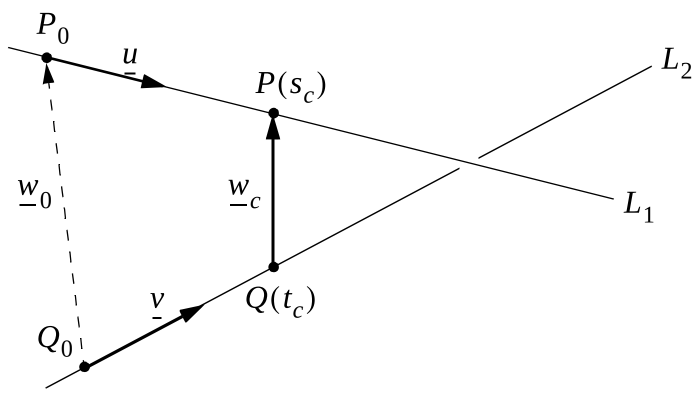


Figure 124 - Closest points on two lines.

8. Appendix B - Unpublished Material on Pinballs

The following unpublished paper gives more details on the hierarchic pinball contact-impact algorithm with Lagrange Multipliers. The paper is incomplete: Sections 8 (Extension to domain decomposition and to spatial partitioning) and 10 (Summary and conclusions) are missing, and Section 9 (Numerical examples) is incomplete.

BLANK PAGE

To be submitted for publication in:
(TO BE DECIDED)

Contact-impact in explicit fast transient dynamics by the hierarchic pinball method with Lagrange multipliers

Folco Casadei[1], Vincent Faucher[2]

[1] *Joint Research Center, Institute for the Protection and Security of the Citizen*
T.P. 480, I-21020 Ispra, Italy
[2] *Commissariat à l'Energie Atomique, Saclay, France*
e-mail: folco.casadei@jrc.it, vincent.faucher@cea.fr

SUMMARY

The pinball algorithm first proposed two decades ago by Belytschko and co-workers is a simple, general and robust method for contact detection. If needed, spatial accuracy may be increased almost at will by splitting the element-based parent pinballs into a hierarchy of descendents. However, when using an implicit Lagrange multiplier technique to enforce impenetrability, redundant constraints are generated in situations of smooth or flat contact and the problem becomes ill-conditioned. A general implementation of the hierarchic pinball method is presented and techniques are proposed to improve geometrical aspects of contact enforcement, while at the same time removing the above mentioned redundancies. Extension to parallel domain decomposition and to spatial partitioning is also considered, and several numerical examples are shown.

KEY WORDS**:** Contact-impact, Pinball algorithm, Explicit, Fast Transient Dynamics.

1 INTRODUCTION

Contact-impact phenomena between solids occur quite often in engineering applications, ranging from metal forming, to automobile crash, to hyper-velocity impact with perforation. Although all these cases involve the notion of contact, the dominating phenomena are quite different physically. In the first case loading is slow, contact is typically smooth, deformations are relatively moderate and friction usually plays a very important role, while in the latter case loading is much faster, reaching complete failure of some parts of the body (fragmentation). As a matter of fact, the range of contact problems is so wide that a variety of approaches and of corresponding algorithms has been developed in the literature to cover the different cases.

In this paper, we focus in particular on transient dynamic applications such as crashes, impacts and explosions occurring as a consequence of natural disasters or of deliberate actions such as terrorist attacks. Loading is typically fast or very fast, friction is usually not a major issue and plastic deformations range from moderate to severe, sometimes up to complete failure and fragmentation of some structural components. Both so-called self-contact (Fig. 1) and body-to-body contact (Fig. 2) are considered.

For this class of problems Finite Element spatial discretization with explicit time integration is typically used. Two major strategies have been proposed in the literature to deal with the contact phenomena occurring in this context: the so-called slide line (2D) or slide surface (3D) algorithms pioneered by Hallquist, Benson *et al.* [1-2], and the so-called pinball algorithm first proposed by Belytschko and co-workers [3-7].

Both algorithms consist of two phases: a *contact detection* (or penetration check) phase, which involves mostly geometrical calculations, followed by a *contact enforcement* phase whereby suitable contact forces are computed to prevent (further) interpenetration. The first phase is radically different in the two approaches: slide surface algorithms use so-called “slave” nodes and “master” surfaces, while the pinball approach uses simple spheres embedded in the elements. The second phase is quite similar if not identical in the two methods. In fact, a variety of techniques may be used in both cases to impose the contact constraints: e.g penalty methods, Lagrangian multipliers, or augmented Lagrangian forms.

In general, slide surface algorithms are quite adequate for smooth contact problems, e.g. in metal forming applications. However, in geometrically more complex situations such as crash or perforation the complexity of geometrical calculations grows rapidly and several ambiguous penetration detection cases may occur, especially in the presence of thin shells. In reference [7] Belytschko illustrates some examples (see Fig. 3) and some others are shown by Winkelmuller [8], see Fig. 4. Because only interpenetration between nodes and element faces are checked, some surface-to-surface or edge-to-surface contacts cannot be detected, not to talk about contact between thin (1D) structural members such as beams and rods. Since in complex simulations the edge-to-surface contacts which can develop are not obvious, several simulations may be needed before all contacts are correctly identified. Solution of industrial problems thus often requires a trial-and-error process.

The pinball method, proposed by Belytschko [6] to overcome all the mentioned shortcomings and originally targeted mostly at impact problems with perforation, is attractive because it considerably simplifies geometric contact detection, reducing it to simple interpenetration checks between couples of spheres (pinballs). By using a pinball splitting technique [7] to better fit smaller and smaller pinballs within slender or distorted elements, the accuracy of contact detection may be increased potentially at will. The resulting method, as far as concerns the contact detection phase, is extremely robust because all the ambiguous cases mentioned above are avoided. Furthermore, a single and unified approach may be applied—with obvious benefits in terms of generality, robustness and computational speed—to all types of finite elements used in realistic discretizations, ranging from bulky continuum-like elements to structural members (shells, beams, bars) and even to material points (e.g. for coupling between FE and particle-based methods).

Yet another advantage of the pinball algorithm which, perhaps somewhat strangely, is not emphasized by Belytschko in the mentioned references is the fact that, unlike the slide surface technique, the algorithm is inherently symmetric. There is no need, at least in principle, to artificially distinguish between a master and a slave body, since pinballs are embedded in both bodies in exactly the same way. Thus input data is substantially simpler and safer and, most importantly, a unique solution for the contact problem is obtained. This is not always the case with sliding surfaces, where the results change, sometimes even considerably, if the definition of master and slave bodies is swapped.

For the above reasons, especially for its generality and robustness in contact detection, the pinball method is an ideal candidate for implementation in a fast transient explicit code, for the treatment of a large variety of contact applications. This has been realized in the EUROPLEXUS code, developed in collaboration by the Joint Research Center (JRC) of the European Commission and by the French Commissariat à l’Energie Atomique (CEA). The main method chosen for contact enforcement is based upon an implicit treatment via Lagrange multipliers, but recently also a fully explicit version with penalty is under testing. Several difficulties have emerged in the implementation via Lagrange multipliers, mainly related to the presence of flat or smooth contacts. The nature of these difficulties is examined in detail below and solutions are proposed, suggested by the comparison with slide surface algorithms, which behave better in these particular situations.

The paper is organized as follows. Section 2 briefly recalls the basic equations of dynamics and the transient explicit solution scheme, into which the contact algorithm is embedded. Section 3 summarizes the chosen Lagrange multipliers approach. The geometrical aspects of contact detection via a hierarchic pinball method are detailed in Section 4. Section 5 deals with the treatment of rebound and Section 6 with the elimination of redundant constraints, which appear to be an inevitable consequence of hierarchic pinball methods. Section 7 briefly discusses the implications of contact over the choice of time integration steps and the optimization of contact search. Section 8 presents the extension of the contact algorithm to domain decomposition and spatial partitioning. Finally, some conclusions are given in Section 9.

All the calculations presented in this work were performed by the EUROPLEXUS code on a Pentium 4 PC with a 3.0 GHz processor and 1GB of RAM.

2 TRANSIENT EXPLICIT SOLUTION SCHEME

Before introducing the contact algorithm, it is useful to briefly recall the governing equations at the base of the transient explicit formulation. We assume for simplicity a Lagrangian description, suitable for the treatment of purely structural applications.

2.1 Governing equation

The governing equation is the conservation of momentum. By expressing equilibrium in the current configuration and by introducing a spatial semi-discretization based on Finite Elements, the following set of discrete differential equations in time may be obtained (bold-face symbols indicate non-scalar quantities):

$$\mathbf{M}\mathbf{a} = \mathbf{F}^{\text{ext}} - \mathbf{F}^{\text{int}}, \tag{1}$$

where $\mathbf{M}$ is the mass matrix, $\mathbf{a}$ is the vector of nodal accelerations, $\mathbf{F}^{\text{ext}}$ are the external forces and $\mathbf{F}^{\text{int}}$ are the internal forces, which may be evaluated by spatial integration over the elements as:

$$\mathbf{F}^{\text{int}} = \sum_e \int_{V_e} \mathbf{B}^T \boldsymbol{\sigma} dV . \tag{2}$$

In (2) the summation symbol represents the ordinary assembly operator over all elements e of the mesh, V_e is the element volume in the current configuration, $\mathbf{B}$ is the matrix of shape function derivatives, of which a superposed T indicates the transpose, and $\boldsymbol{\sigma}$ is the Cauchy or “true” stress tensor. The nodal accelerations are formally obtained from (1) as:

$$\mathbf{a} = \mathbf{M}^{-1}(\mathbf{F}^{\text{ext}} - \mathbf{F}^{\text{int}}) . \tag{3}$$

However, since the mass matrix $\mathbf{M}$ may be lumped (i.e., reduced to diagonal form), no matrix inversion or system solution is actually required and (3) may be simply treated by considering each degree of freedom (dof) j separately:

$$a_j = (F_j^{\text{ext}} - F_j^{\text{int}}) / m_j . \tag{4}$$

2.2 Explicit time integration

Time integration of (1) under the form (4) is achieved via the so-called central difference (CD) scheme, which is implemented as follows in a typical explicit code. Assume that a complete solution, i.e. all discretized quantities, are known at time t^n. First, an intermediate velocity at mid-step is introduced:

$$\mathbf{v}^{n+1/2} = \mathbf{v}^n + (\Delta t/2)\mathbf{a}^n . \tag{5}$$

This is the *constant* velocity that would transform configuration n into $n+1$ over a time interval $\Delta t = t^{n+1} - t^n$. Then the new displacements are given by:

$$\mathbf{d}^{n+1} = \mathbf{d}^n + \Delta t \mathbf{v}^{n+1/2} . \tag{6}$$

On the new (i.e., the current) configuration induced by these displacements:

$$\mathbf{x}^{n+1} = \mathbf{x}^n + \Delta t \mathbf{v}^{n+1/2} = \mathbf{x}^0 + \mathbf{d}^{n+1} , \tag{7}$$

the internal forces can be evaluated via eq. (2) by applying the material constitutive relations. Then, the new accelerations $\mathbf{a}^{n+1}$ can be directly computed via the discretized equilibrium equations (1) under the form (4), and finally the new velocities $\mathbf{v}^{n+1}$ are obtained from:

$$\mathbf{v}^{n+1} = \mathbf{v}^n + (\Delta t/2)(\mathbf{a}^n + \mathbf{a}^{n+1}) \tag{8}$$

The CD time integration scheme is explicit in that all quantities in the right-hand-side terms are known when the equations are applied. Thus, no system solver is needed except possibly for the enforcement of essential boundary conditions, in case an implicit treatment via Lagrange multipliers is chosen.

3 TREATMENT OF CONSTRAINTS

Most essential boundary conditions may be formulated as linear constraints of the form:

$$\mathbf{C}\boldsymbol{\upsilon} = \mathbf{b} , \tag{9}$$

where $\mathbf{C}$ is a matrix of known coefficients, $\boldsymbol{\upsilon}$ is the vector (subset) of linked degrees of freedom (dofs), typically the velocities, and $\mathbf{b}$ is a known vector. In general, both $\mathbf{C}$ and $\mathbf{b}$ may be function of time. A variety of methods may be used to impose such constraints, among which perhaps the most popular are the penalty method (explicit) and the Lagrange multipliers method (implicit).

In the EUROPLEXUS code Lagrange multipliers have been chosen by CEA [9] as the default technique because of their generality. The implicit formulation ensures proper combination of all constraints and unique solutions, irrespective of the order in which the constraints are given. There are in fact no specific input parameters to be “tuned”, unlike in the penalty method. This simplifies also input preparation and input error detection. In fact, if the imposed conditions are physically incompatible (as may be not readily evident in large complex applications) the resulting linear system becomes singular and the problem is promptly detected. The price to be paid is the computational cost of the implicit solution of a linear system in an otherwise fully explicit environment. However, the CPU overhead with respect to an explicit technique is usually negligible, if the subset of constrained dofs is relatively small.

3.1 Lagrange multipliers

With reference to the time integration scheme of Section 2.2, suppose that a configuration $n+1$ at t^{n+1} has been reached. The velocity and acceleration corresponding to this configuration are not known yet, but the internal forces and the external loads (*natural* boundary conditions) are known since they depend only upon the current configuration and upon time. For simplicity, all quantities expressed at time t^{n+1}, i.e. in the current configuration, are indicated without the superscript $n+1$ in the following discussion.

Consider the subset of degrees of freedom for which *essential* boundary conditions are imposed. The equilibrium equations for this subset can be written, in analogy with eq. (1):

$$\mathbf{m}\boldsymbol{\alpha} = \mathbf{f}_{\text{ext}} - \mathbf{f}_{\text{int}} + \mathbf{r} = \mathbf{f} + \mathbf{r}\,, \tag{10}$$

where $\mathbf{m}$ is the mass matrix, $\boldsymbol{\alpha}$ is the vector of accelerations, $\mathbf{f}_{\text{ext}}$ and $\mathbf{f}_{\text{int}}$ are the vectors of external loads and of internal forces, respectively, $\mathbf{r}$ are the unknown reactions and we have posed $\mathbf{f} \equiv \mathbf{f}_{\text{ext}} - \mathbf{f}_{\text{int}}$ for simplicity. Note that similar, but distinct, symbols have been used here with respect to eq. (1)—e.g. $\mathbf{m}$ instead of $\mathbf{M}$ for the mass matrix, etc.—to stress the fact that these equations involve just a (usually small) *subset* of the degrees of freedom in eq. (1).

Now assume that essential boundary conditions are expressed by a linear set of constraints of the form (9) on the velocities or, more precisely, on the next mid-step ($t^{n+3/2}$) velocities:

$$\mathbf{C}\boldsymbol{\upsilon}^{n+3/2} = \mathbf{b}\,. \tag{11}$$

To solve equation (10) for the accelerations $\boldsymbol{\alpha}$, one should first determine the unknown reaction forces $\mathbf{r}$. To this end, use is made of Lagrange multipliers associated with the constraint (11). *Without loss of generality*, the unknown reactions can in fact be expressed as:

$$\mathbf{r} = \mathbf{C}^T\boldsymbol{\lambda}\,, \tag{12}$$

where $\boldsymbol{\lambda}$ is the vector of Lagrange multipliers. Substituting into (10) yields:

$$\mathbf{m}\boldsymbol{\alpha} = \mathbf{f} + \mathbf{C}^T\boldsymbol{\lambda}\,. \tag{13}$$

Before introducing into this equation the constraint (11), which is based upon the velocities, one should first transform (11) into the equivalent form expressed on the accelerations. This is achieved by exploiting the time integration scheme described in Section 2.2.

The central difference scheme for the velocity, see eqs. (5) and (8), gives:

$$\boldsymbol{\upsilon}^{n+3/2} = \boldsymbol{\upsilon}^{n+1/2} + (\Delta t^n + \Delta t^{n+1})\boldsymbol{\alpha}/2 = \boldsymbol{\upsilon}^{n+1/2} + \gamma\boldsymbol{\alpha}\,, \tag{14}$$

where again a different symbol $\boldsymbol{\upsilon}$ instead of $\mathbf{v}$ is used for the subset of linked dofs and γ indicates the (known) coefficient $\gamma \equiv (\Delta t^n + \Delta t^{n+1})/2$. With this the constraint (11) becomes:

$$\mathbf{C}\boldsymbol{\upsilon}^{n+1/2} + \mathbf{C}\gamma\boldsymbol{\alpha} = \mathbf{b} \qquad \text{or} \qquad \mathbf{C}\gamma\boldsymbol{\alpha} = \mathbf{b} - \mathbf{C}\boldsymbol{\upsilon}^{n+1/2}\,. \tag{15}$$

Equations (15) may be interpreted as equivalent forms of the constraint (11), expressed on the accelerations rather than on the velocities. Note in fact that the old velocities $\boldsymbol{\upsilon}^{n+1/2}$ are known and therefore the right-hand side of the second of (15) is a known vector. Multiplying the equilibrium equation in the form (13) by $\mathbf{C}\gamma\mathbf{m}^{-1}$ and rearranging gives:

$$\mathbf{C}\gamma\mathbf{m}^{-1}\mathbf{C}^T\boldsymbol{\lambda} = \mathbf{C}\gamma\boldsymbol{\alpha} - \mathbf{C}\gamma\mathbf{m}^{-1}\mathbf{f} \qquad \text{or} \qquad \mathbf{D}\boldsymbol{\lambda} = \mathbf{w}\,. \tag{16}$$

By solving this linear system one finds the Lagrange multipliers $\boldsymbol{\lambda}$, and then the reactions $\mathbf{r}$ are obtained from (12). Finally, the accelerations $\boldsymbol{\alpha}$ along the constrained dofs are explicitly computed from (10)—in fact recall from Section 2.2 that the mass matrix $\mathbf{m}$ is lumped—and the time integration procedure may go on.

In eqs. (16) the quantity $\mathbf{D}$ is a square symmetric matrix, called the *matrix of connections*, and $\mathbf{w}$ is a vector. Both $\mathbf{D}$ and $\mathbf{w}$ are known, as appears from their definitions, but in general they must be evaluated at each step, since the coefficients $\mathbf{C}$ and $\mathbf{b}$ usually vary with time.

Remark 1. One might wonder why it is chosen to impose the constraint (11) on the mid-step velocity $\boldsymbol{\upsilon}^{n+3/2}$ rather than, say, on the full-step velocity $\boldsymbol{\upsilon}^{n+1}$:

$$\mathbf{C}\boldsymbol{\upsilon}^{n+1} = \mathbf{b}. \tag{17}$$

The reason is that the fundamental quantity of the CD time integration scheme is the mid-step velocity as given by (5). Note also, incidentally, that using the full-step velocity would be inconsistent at the zero-step (initial time) of the transient calculation i.e. for $\boldsymbol{\upsilon}^{n+1} = \boldsymbol{\upsilon}^{0}$. Initial velocities $\boldsymbol{\upsilon}^{0}$ may in fact be freely prescribed and should not be modified by the time integration algorithm. Results are usually quite similar for permanent constraints such as node blockages. However, as shown in Fig. 8 for a simple academic 1D bar impact test case, the algorithm based upon eq. (11) gives much smoother numerical results than eq. (17) in the presence of non-permanent (e.g. contact) constraints.

4 CONTACT DETECTION BY PINBALLS

The pinball algorithm in its basic form was first introduced by Belytschko and Neal in references [4] and [6]. The major goal with respect to the more conventional slide line and surface techniques was to simplify interpenetration checks, eliminate many conditional branches and thus provide a readily vectorizable procedure, well suited for some computer architectures in use at that time. The original target was the simulation of high-speed impact and penetration phenomena, in conjunction with suitable element erosion procedures.

4.1 Basic pinball method

According to [6], the main idea of the pinball algorithm is to enforce the impenetrability condition and to define the interpenetration via a set of spheres, or pinballs, embedded in the finite elements (just one pinball per element) as shown in Fig. 6a. Interpenetration between the contacting bodies is approximated by interpenetration between the pinballs.

In [6] the pinball centre is the average of its element's nodes while the radius is such that the pinball volume equals the element volume (*equivalent* radius). Alternatively, one might use a larger radius, encompassing all the element's nodes (*encompassing* radius). While the center is evaluated at every time step, the radius is kept constant, thus assuming that element deformation is not too large and occurs (plastically) at constant volume.

If R_1, R_2 are the radii of two pinballs, and d_{12} the distance between their centers $\mathbf{C}_1$, $\mathbf{C}_2$, then contact (or more precisely interpenetration) has occurred when:

$$d_{12} = \left\| (\mathbf{C}_1 - \mathbf{C}_2) \right\| < R_1 + R_2 \tag{18}$$

where $\| \ \|$ designates the length of a vector.

Once detected the contact, one must introduce suitable contact forces preventing (further) interpenetration of the contacting bodies. Ref. [6] presents two implementations, one based upon penalty methods, which is also retained in subsequent work [7], and the other based on Lagrange multipliers. Whatever implementation is chosen, contact forces should ensure that:

$$\left(\mathbf{v}_A - \mathbf{v}_B\right) \cdot \hat{\mathbf{n}} \le 0 \tag{19}$$

where $\mathbf{v}_A$, $\mathbf{v}_B$ are the velocities of the two bodies at some *contact point(s)*, and $\hat{\mathbf{n}}$ represents a suitable *normal direction* to the contact surface (hats stand for unit vectors). The inequality sign accounts for possible rebound (see below). To complete the algorithm one must choose how to compute the contact point(s), i.e. how to express $\mathbf{v}_A$, $\mathbf{v}_B$ in terms of the nodal velocities of the elements containing the impacting pinballs, and how to compute the normal $\hat{\mathbf{n}}$.

In ref. [6] two alternatives are considered for the contact point(s): either the pinballs centres (corresponding to the associated elements' centroids), or the centres of the contacting element faces. In the first case, the contact force is equally distributed over the 8 nodes of the hexahedral elements used in [6] while in the second case the force is equally distributed over the 4 nodes of the contacting face. How to determine the contacting face is not specified.

As concerns the normal $\hat{\mathbf{n}}$, the following expression is proposed in [6]:

$$\hat{\mathbf{n}} = (\hat{\mathbf{n}}_2 - \hat{\mathbf{n}}_1) / \|\hat{\mathbf{n}}_2 - \hat{\mathbf{n}}_1\|, \tag{20}$$

where $\hat{\mathbf{n}}_1$ and $\hat{\mathbf{n}}_2$ are "the" normals associated with the two contacting pinballs. Thus, in this approach each pinball must have a unique, well-defined normal. According to [6] this normal is defined by the so-called assembled surface normal algorithm of Belytschko and Lin [3], which assembles an approximate normal to outside surfaces. The normal is non-zero only on outside surface nodes, and pinballs are placed only in elements with at least one node with a non-zero normal, see Fig. 6b.

Unfortunately, it is not specified how to pass from the assembled normals at nodes to the normals at pinballs, which are the ones needed in expression (20). Maybe simple averaging was used in [6] and it is possible that the actual expression adopted is not too important for the perforation calculations with erosion considered in that reference. As observed by the authors, in any case the piecewise spherical contact surface resulting from pinballs is much smoother than the piecewise linear irregular surface resulting from element erosion.

Although eq. (20) works relatively well for bulky continuum-like bodies, the same may not be said for thin shell or beam elements. As a matter of fact, in the hierarchic pinball algorithm [7], which deals also with shell elements, Belytschko and Yeh use a simpler approach: the contact force (they use a penalty approach) is always exerted in the direction joining the pinball (or descendent pinball, in case of hierarchy) centers, i.e.:

$$\hat{\mathbf{n}} = \left(\mathbf{C}_1 - \mathbf{C}_2\right) / \|\mathbf{C}_1 - \mathbf{C}_2\| = \left(\mathbf{C}_1 - \mathbf{C}_2\right) / d_{12}. \tag{21}$$

Expressions (20) or (21) are adequate for problems where sliding and friction are not crucial, such as penetration and crashworthiness. However, in the "smooth" sliding of two relatively flat bodies, spurious oscillations may occur in the contact normal and thus in the direction of contact forces, due to the pinballs curvature. These effects were not studied in [6].

Contact detection by the basic pinball algorithm is extremely robust (conservative) and avoids all pathological situations discussed in Section 1. However, spatial resolution is clearly insufficient in cases of large practical importance (see Fig. 7), i.e.: a) with very irregular or distorted (as a consequence of large deformation) continuum elements; and b) with beam, plate or shell elements having small (or zero) topological thickness.

4.2 Hierarchic pinball method

Belytschko and co-workers recognized this deficiency and proposed an improved algorithm, based on a (hierarchic) splitting procedure, see [7]. The idea is quite simple: to improve spatial resolution of contact, a hierarchy of pinballs is constructed. A unique pinball is first and constantly associated with each element like in the basic algorithm. These are the so-called *parent*, or 0-level pinballs. Whenever two such pinballs are found to interpenetrate, they are (recursively) split into smaller (higher-level) *descendent* pinballs, which are further checked for penetration until either penetration is no longer detected or some prescribed maximum level (or minimum size) of these descendent pinballs is reached. Normally the size of pinballs is roughly halved at each splitting operation.

Thus the interpenetration of parent pinballs becomes a necessary, but not sufficient, condition for contact. It might therefore be safe and convenient to take both the parent pinballs and the descendents up to the fore-last level of the hierarchy as large as needed to ensure detection of all contacts (“encompassing” radius as described in Section 4.1). For the last level of the hierarchy an “equivalent” radius in the sense of Section 4.1 is probably a better choice.

Fig. 8a illustrates an example of hierarchic pinball generation in 2D for a regular quadrilateral element. The parent (zero-level) pinball has the same volume as the element. This pinball is subdivided into 1-level pinballs, each encompassing one fourth of the original element (represented by the dotted lines). These are further subdivided into 2-level pinballs and so on. Note that only “external” descendent pinballs, i.e. belonging to the external parts (faces F_2, F_3), of the element, need be generated, because interpenetration necessarily initiates in these parts of the contacting bodies. In addition, sub-pinballs are (recursively) generated only from pinballs that are found to be actually in contact (in Fig. 8 we assume that all “outer” pinballs are contacting an external body, not shown in the figure for simplicity).

A compromise should be found between two opposite requirements: pinballs should be kept small to increase spatial resolution of contact detection, but not too small, in order to limit the reduction of explicit integration time steps (see Section 7). Fig. 8b illustrates the hierarchic pinball splitting for a shell element with zero topological thickness, an application of large practical importance. In this case there is a physical criterion which indicates (or rather limits) the minimum size of descendent pinballs and thus the maximum depth of the hierarchy: the final pinball diameter should be of the same order as the physical thickness of the associated elements. This accounts automatically—within the contact algorithm itself—for the shell thickness which is not represented topologically, a difficult task with slide surface contact-impact methods. Fig. 8c illustrates the hierarchic pinball splitting for a 3D hexahedron element. One may proceed similarly for the other element shapes. In the present work, the whole family of linear-displacement finite elements is considered.

4.3 Implementation

The hierarchic pinball algorithm has been implemented in the explicit FE code. The basic algorithm can be obtained as a special case, by specifying zero depth of the hierarchy. By default, contact enforcement is achieved by the implicit Lagrange multiplier technique of Section 3.1, applied to the constraints (19). An alternative fully explicit penalty-based implementation is under testing.

5 TREATMENT OF REBOUND

In contact-impact one must deal with rebound, i.e. with the unilateral nature of contact conditions. The contacting bodies must be free to detach whenever appropriate conditions are met. In impact problems, rebound occurs when a tensile stress wave—e.g. generated by reflection of the main compressive impact wave at a free boundary—reaches the contact interface.

Consider first the simple case of a single contact constraint resulting e.g. from localized contact between two bodies. A possible strategy is to provisionally impose the constraint (19) with the equals sign, i.e. as a bilateral constraint, compute the corresponding Lagrange multiplier and reaction force, and then retain this force only if it tends to keep the two bodies apart. This technique, that one might denote *a posteriori* treatment of rebound, works well e.g. with simple node-to-node contact algorithms, because these indeed produce mutually independent constraints. However, it is not adequate in general with slide surface techniques and perhaps even more with hierarchic pinball methods, because the resulting constraints are (sometimes heavily) coupled, e.g. in situations of flat contact.

Thus, in the general case the problem of rebound would in principle require a fully implicit (e.g. iterative) solution. However, this is hard to justify in the chosen computational scheme which, as shown above, is fully explicit except for the solution of the constraints system (16). Therefore, an approximate explicit *a priori* treatment of rebound is adopted. As an indication of incipient rebound we consider the sign of the rate of change $\dot{\delta}$ of the (oriented) distance δ between centers of contacting (sub-)pinballs A and B, see Fig. 9. This is defined as:

$$\delta = (\mathbf{B} - \mathbf{A}) \cdot \mathbf{n}_{AB} \quad \text{with} \quad \mathbf{n}_{AB} = (\mathbf{B} - \mathbf{A}) / \|\mathbf{B} - \mathbf{A}\| \tag{22}$$

and is a positive scalar in the current configuration, i.e. time station $n+1$ in Fig. 9a. To estimate the rate of change of δ we compute the "free" positions of pinball centers at the next time station $n+2$, i.e. by neglecting the pinball contact forces (and any other constraints):

$$\mathbf{A}^* = \mathbf{A} + \mathbf{v}_A^* \Delta t = \mathbf{A} + \mathbf{v}_A^{n+1/2} \Delta t + \mathbf{a}_A^* (\Delta t)^2 \tag{23}$$

where $\mathbf{v}_A^*$ is the "free" velocity at $n+3/2$ and $\mathbf{a}_A^*$ the "free" acceleration at $n+1$, which can be computed via eq. (3) or (4) by neglecting all constraints. Thus in analogy with (22) the "free" inter-centers distance at $n+2$ would be:

$$\delta^* = (\mathbf{B}^* - \mathbf{A}^*) \cdot \mathbf{n}_{AB}^* \quad \text{with} \quad \mathbf{n}_{AB}^* = (\mathbf{B}^* - \mathbf{A}^*) / \|\mathbf{B}^* - \mathbf{A}^*\| \tag{24}$$

This is again a positive scalar quantity, which may not be directly compared with δ because the orientation of $\mathbf{n}_{AB}^*$ is different in general from that of $\mathbf{n}_{AB}$, see Fig. 9. Therefore, we compute first the scalar product $s = \mathbf{n}_{AB}^* \cdot \mathbf{n}_{AB}$, and then:

$$\delta^{**} = \delta^* \operatorname{sign}(s) \tag{25}$$

This guarantees that, in case the A pinball would "overtake" the B pinball in the next free configuration, a consistently signed (negative) value is obtained for δ^{**}. Finally, the rate of change of inter-centers distance is estimated by:

$$\dot{\delta} = (\delta^{**} - \delta) / \Delta t \tag{26}$$

The pinball contact constraint under consideration is retained only if $\dot{\delta} < 0$.

6 ELIMINATION OF REDUNDANT CONSTRAINTS

One major difficulty that has emerged during the implementation of the hierarchic pinball method is the onset of *redundant contact conditions*. This fact had not been investigated (nor mentioned) in [7], possibly because Belytschko and Yeh adopt a penalty method to compute the contact forces. However, in conjunction with the Lagrange multipliers method chosen here, the presence of redundant constraints renders the connections matrix $\mathbf{D}$ of (16) singular, and thus the solution of the system becomes impossible or at least more laborious.

For example, consider Fig. 10a. Two elements A and B come into "flat" contact, i.e. along a whole side. At level 0 (parent pinballs) there is just one contact constraint, which may be written e.g. by linking the velocities at pinball centers (such constraints are represented by thick segments in Figs. 10 to 13). Passing to level 1, we obtain two constraints. In both cases, no redundancies are observed. However, at level 2 of the hierarchy four constraints would be obtained, of which two are redundant, i.e. the $\mathbf{D}$ matrix is twice singular. The figure presents also the case corresponding to an intermediate level 1.5 whereby three constraints would be obtained, of which one is redundant. This case is fictitious because the actual hierarchic splitting procedure generates only "integer" levels, but it is useful for illustration purposes.

A qualitative explanation of redundancies is that when several contacts are detected between descendent pinballs sharing the same couple of parents, i.e. associated with the same couple of finite elements, the corresponding constraints involve the same set of nodes, i.e. the same degrees of freedom. Intuitively, only a limited number of constraints may be written independently, more precisely one constraint for each node of a contacting face. This means at most two constraints in 2D, and three or four constraints in 3D, when using linear-displacement continuum FE. Any extra constraints are redundant and render the matrix singular. This explains why redundancies are observed neither with the basic pinball method, nor with slide surface techniques. In the former case, there is at most one constraint between each FE couple, while in the latter case at most one constraint is written for each slave node.

Fig. 10b illustrates another example of redundant constraints, due to flat contact between two neighboring element couples (this may be generalized to the case of two long flat bodies in smooth contact). Here we obtain one redundancy already at hierarchy level 1, due to neighboring elements contributions to the common contacting nodes.

6.1 The "Common Normal" algorithm

It is of course desirable to eliminate redundant constraints *before* attempting to solve the system (16), both for efficiency and for accuracy reasons. Many redundancies may be removed by applying the criterion mentioned above which limits the number of constraints to be retained for each contacting element couple. Fig. 11 illustrates the process in the simple 2D case of Fig. 10a. Assuming a hierarchy level 1.5 (just one redundant constraint) for simplicity, there are three alternative ways of removing the singularity, as sketched in the figure, i.e. by removing the right, the central or the left constraint. Analytically, results are indeed equivalent. However, numerically it is easily verified that removing the central constraint is the best solution, since the resulting constraints matrix is better conditioned. Note that, however, this technique does not remove all redundancies: e.g., it has no effect in the case of Fig. 10b.

The search for redundant constraints according to the technique described above is a computational burden specific to hierarchic pinballs, but it may be exploited to improve model behavior in situations such as those depicted in Fig. 12. Case 1 illustrates the perfect, ideal flat contact between two elements with level 2 pinballs. Of the four detected contacts two are redundant, so the above algorithm would retain only the two "external" contacts as indicated by the thick segments. Assuming that contact constraints are written along the lines joining the pinballs centres, one would obtain the two normal directions $\hat{\mathbf{n}}_1$ and $\hat{\mathbf{n}}_4$, which in this case are coincident and perpendicular to the interface between the two bodies.

However, when any imperfections are present as shown in Cases 2 to 4 in Fig. 12, due e.g. to misalignment, size mismatch or deformation, one would end up with normals that are no longer coincident (Cases 3 and 4) and anyway are not perpendicular to what may be considered as the "common interface" of the two contacting bodies. This is contrary to physical intuition, and the resulting contact force components tangential to the interface, although usually small, would produce spurious sticking or friction-like effects that act against the free relative sliding of the two bodies (inviscid contact is assumed here). For large relative sliding, an oscillating spurious tangential force would result, qualitatively indicated in the inset of Fig. 13, whose amplitude and period would depend on the size of the contacting pinballs.

In all four cases illustrated in Fig. 12, the desired behavior is that of Case 1, i.e. only the two "external" constraints should be retained, and these should have coincident normals $\hat{\mathbf{n}}_1 \equiv \hat{\mathbf{n}}_4$, both perpendicular to "the" contact interface. This may be obtained by the following "Common Normal" (CN) algorithm (see Fig. 13). Considering the case of 2D continuum elements for simplicity (the 3D situation is analogous but is not presented for brevity) one must accept at most two contacts between descendent pinballs derived from the same couple of parents. If there are more than two, we construct the "mid-points" $\mathbf{M}_i$ defined as the central points of the common zones (grayed in Fig. 13) of segments joining the sub-pinball centers. For each couple of mid-points i, j we compute their distance d_{ij}. We retain only the two extreme contacts I, J such that $d_{IJ} = \max(d_{ij})$. Both contacts are assigned the same (common) normal $\hat{\mathbf{n}}_C$, given by the direction perpendicular to segment $\mathbf{M}_I\mathbf{M}_J$.

If one (at least) of the contacting elements is a material point (1-node particle) or a bar/beam element (2-node segment), the above CN algorithm requires some simple adaptations, not discussed here for brevity. Interested readers may find further details in ref. [11].

6.2 Improving contact points by constraint collapse technique

As mentioned in Section 4.1 and in ref. [6], the pinball method offers a certain freedom as concerns the points at which discrete contact conditions are imposed. In the present implementation, by default, such points are the pinball (or sub-pinball) centers. However, in principle any point of the pinball may be chosen, since contact locations in the discrete model are determined only with a certain approximation, of the order of the (sub-)pinball radius.

When writing down the constraints (19), if the assumed discrete contact points A, B are internal to the parent element, then resulting expressions via element shape functions involve all nodes (dofs) of the element, thus leading to relatively complex and coupled constraints. This is not the case with slide surface contact algorithms, whereby at most one constraint for each node of the slave surface is generated, thus avoiding any redundancies.

To simplify the constraints and to reduce the risk of redundancies in the hierarchic pinball method it seems therefore better, whenever appropriate, to choose discrete contact points that lie on the outer surface of the parent element or that, even better, coincide with a node of the parent. In the first case the constraint involves just the nodes of the contacting face, in the second just the contacting node. In this way, the expressions obtained from pinballs are quite similar to those that would result from node-to-surface contact algorithms.

In the following, we will refer to this technique as constraint "collapsing" on the nodes (or on the element surface), or simply as Nodal Collapse (NC). It is therefore convenient to classify descendent pinballs in three categories, as shown in Fig. 14: corner pinballs (close to nodes), side pinballs (along element sides) or face pinballs (3D only). With this nomenclature, the NC algorithm is as follows:

- *NC Rule 1*. For continuum elements: a corner pinball collapses onto the nearest node, a side pinball collapses onto the nearest point along the side and a face pinball (3D only) collapses onto the nearest point on the face.
- *NC Rule 2*. For 2D/3D beam elements and 2D shells (which have 2 nodes and 2 faces): a corner pinball collapses onto the nearest node, a side pinball does not collapse at all because its center lies already on the side. Furthermore, there are no face pinballs.
- *NC Rule 3*. For 3D shell elements (which have 3 or 4 nodes and 2 faces): a corner pinball collapses onto the nearest node, a side pinball collapses onto the nearest point along the side and a face pinball does not collapse at all because its center lies already on the face.

For example, consider the simple flat contact between two quadrilateral elements depicted in Fig. 15. The left part of the Figure shows the four "raw" contacts detected by a hierarchic pinball method of the second order (level 2 descendents). By applying first the CN algorithm of Section 6.1, as shown in the central part of the Figure, only the two extreme contacts (C_1, C_4) are retained, both acting along the common normal direction $\hat{\mathbf{n}}_c$. Then, according to the NC algorithm, the contact points collapse onto the nodes, see the right part of the Figure, since all involved sub-pinballs are corner pinballs. The normal directions are not affected by this procedure and remain those resulting from the CN algorithm ($\hat{\mathbf{n}}_c$).

The two resulting constraints are thus completely independent from each other, since each of them involves just one node of each element. The resulting constraints coincide in this case with those obtained from a node-to-node contact method, which seems quite appropriate in the present example.

As another example of the beneficial effects of the combined CN/NC algorithms, consider the misaligned flat contact case of Fig. 16. Of the four sub-pinballs which remain after the CN procedure, two collapse on the nodes as before, while the other two collapse on the element side. The resulting constraints are again very similar to those obtained from a slave node / master face technique, the only difference being that the discrete contact points on the two facing elements do not coincide perfectly in general. However, the maximum distance between such points is of the order of the contact resolution (sub-pinball radius) and may therefore, at least in principle, be rendered as small as desired by increasing the hierarchy depth.

6.3 Eliminating residual redundancies

As already mentioned, the CN algorithm of Section 6.1 may eliminate most, but in general not all, the redundancies which result from straightforward application of a hierarchic pinball method in flat contact conditions. Therefore, after application of the CN and of the NC algorithms, a final Residual Constraint (RC) elimination algorithm may be needed.

Consider for example the simple case of 2D flat aligned contact between two couple of elements, shown in Fig. 10b. Assume a level 2 hierarchy, so that there are 8 “raw” contacts between descendent pinballs as indicated in Fig. 17a. After application of the CN algorithm, only the four constraints C_1, C_4, C_5, C_8 remain, see Fig. 17b. All contacting sub-pinballs are corner pinballs, so that the NC algorithm gives the situation of Fig. 17c. It is clear that of the four remaining constraints, one is redundant. More precisely, C_4 and C_5 are the same constraint and thus one of the two must be eliminated. The final result, shown in Fig. 17d, is identical to the one that would result from a node-to-node contact algorithm or from a slide surface algorithm: three node-to-node constraints remain, all having the same normal in this simple ideal case.

Identification of the redundancy is simple in this case because the two constraints C_4 and C_5 are of the same type (node-to-node constraints) and involve the same couple of nodes. Of course, in geometrically more complex situations and especially in 3D detecting all the redundancies may become a more complicated task. An empirical procedure (RC algorithm) has been set up which treats all possible cases by first subdividing the constraints into groups of node-to-node constraints (like in the simple example considered), or node-to-point constraints, or point-to-point constraints. The full procedure may not be listed here for brevity, but interested readers may find all details in ref. [11].

6.4 Résumé of the proposed hierarchic pinball algorithm

It is perhaps worthwhile to give a short résumé of the general-purpose hierarchic pinball algorithm. It consists of the following phases:

- Detect all “raw” contacts between sub-pinballs, as described in Section 4.2.
- Apply the common normal (CN) algorithm of Section 6.1. This produces “better” normals in flat-like contacts and eliminates most redundancies.
- Collapse each constraint onto the nearest node (NC), side or face, as described in Section 6.2. This simplifies the form of the constraints, rendering them more similar to those that

would result from more traditional slave node / master surface algorithms. It also facilitates the following final removal of surviving redundancies.

- Eliminate any surviving redundant constraints (RC), as described in Section 6.3.
- Apply the a priori rebound detection algorithm, as described in Section 5, and reject any rebounding constraints.
- Add the remaining contact constraints to the other links, solve the linear system and compute the contact forces, as described in Section 3.1.

The algorithm requires no tuning parameters, unlike e.g. penalty-based methods, and has been applied successfully in a large variety of contact situations. Since the elimination of redundant constraints is a relatively expensive operation, the user may decide whether or not to activate it. This is actually necessary only in cases where flat contact may occur.

7 STEP CONTROL AND CONTACT SEARCH OPTIMIZATION

In explicit schemes such as the one adopted here the size of the time increment Δt is governed by the stability (Courant) condition:

$$\Delta t = \varphi \Delta t^{\text{stab}} = \varphi \Delta x / c \tag{27}$$

where Δt^{stab} is the stability step which, in sub-sonic problems, may be estimated as the smallest element length Δx divided by the sound speed c, and φ is a safety factor ($\varphi < 1$).

The pinball algorithm introduces additional constraints on Δt. In fact, for the method to work properly, pinball penetration must be detected before it exceeds a limit value, namely before the centers of the two impacting pinballs lie on opposite sides with respect to their positions just before impact. Consider the example in Fig. 18a. Let $\mathbf{v}_A$, $\mathbf{v}_B$ represent the pinball velocities just before contact, say at time station $n+1$. Then, if the chosen time increment $\Delta t_2 = t^{n+2} - t^{n+1}$ is too large, at the following time station ($n+2$) one might obtain the situation shown in Fig. 18b: the centre of pinball A has “overtaken” the centre of pinball B with respect to the positions they occupied at the previous step. In other words, the pinballs unit normal $\hat{\mathbf{n}}_{AB}$ (oriented segment joining the two centres) has “changed sign” in just one step. In configuration $n+1$ no penetration occurs and thus no contact forces are generated. In configuration $n+2$ there is penetration, but the two pinballs appear to be detaching from each other (apparent rebound) so that, again, no contact forces are generated. In conclusion, pinball A “passes” undisturbed across pinball B.

This situation is unlikely to occur, at least in sub-sonic problems, in the basic pinball algorithm because pinballs are relatively large. However, it becomes more and more likely in hierarchic methods as the size of descendents decreases, unless appropriate measures are taken. Such time step limitations are needed not only with pinballs, but also in slide surface algorithms. The advantage of the pinball method is that it offers a natural and effective way of computing step limitations as part of the contact checking algorithm itself. In fact, these may be obtained at little extra cost along with normal penetration checks. For each couple of parent pinballs A, B eq. (18) is checked, i.e. $d_{AB} < (R_A + R_B)$. At the same time, we compute the (oriented) relative velocity of the two pinballs:

$$v_{AB} = (\mathbf{v}_A - \mathbf{v}_B) \cdot \hat{\mathbf{n}}_{AB} \tag{28}$$

and then, if this quantity is positive ($v_{AB} > 0$, i.e. the two pinballs are approaching each other) we impose the following limitation over the time increments of both elements A and B:

$$\Delta t_2 = t^{n+2} - t^{n+1} < \left(d_{AB} / v_{AB} \right) \tag{29}$$

Keeping the pinballs relatively large in size (encompassing radius) down to the fore-last of hierarchy level ensures not only safe contact capturing, but also early detection of time increment limitations and thus effective and gradual step reduction *before* actual contact occurs.

Another important practical aspect is optimization of contact search operations. Checking each (parent) pinball against each other is an $O(N^2)$ algorithm which becomes prohibitive for large numbers of parent pinballs N. Grouping pinballs into (user-defined) "sets" each representing a separate body helps, because distance checks are skipped for pinball couples belonging to the same set, but this technique may not be applied to self-contact problems. In the present implementation a standard search optimization algorithm based on bucket sorting technique is used. Space is subdivided into a regular grid of cells and pinballs in each cell are checked only against those in the same cell or in direct neighbor cells.

8 EXTENSION TO DOMAIN DECOMPOSITION AND TO SPATIAL PARTITIONING

(To be written).

9 NUMERICAL EXAMPLES

9.1 Cable wrapping

This test involves two elastic cables (see Fig. 19), discretized by 20 and 4 two-node cable-like elements, respectively. These elements react in traction, but not in compression (nor in bending). The shorter cable is fixed at both ends, while the longer cable is fixed at one end and has a 100 kg mass attached at the other end, represented by a material point. The mass has an initial velocity of 100 m/s and this produces the wrapping of the long cable around the short one.

One parent pinball is embedded in each of the cable elements, see Fig. 19a. A hierarchy of level 2 with CN algorithm is used to detect the contacts. First contact occurs at 56 ms and the wrapping process terminates at 833 ms, after 4 complete turns. At this time, due to elastic oscillations of the cables, some rebound (un-winding) starts, see final result at 900 ms. The star-shaped structure appearing in the zoomed-up picture is nothing else than the (coarse) discrete cable elements: obviously a 2-node-element may not "bend" upon itself. This problem shows the versatility of pinball contact detection and would be very difficult if not impossible to treat with node-to-surface techniques (cables have no surface). Another added value of the pinball method is that the contacts themselves (i.e. the contacting descendents and the assumed contact points) are easily visualized for inspection, as shown in the pictures of Fig. 19.

Note that two "bodies" are declared in this problem, each one corresponding to one cable. Self-contact, i.e. contact between (sub-)pinballs belonging to the same body, is not activated. This is why the long cable wraps up by remaining strictly in the $x-z$ plane: the cable may not impact upon itself. Moreover, no pinball is attached to the lumped mass.

9.2 Sphere indentation

This test simulates displacement-driven penetration of a rigid spherical indenter of radius R into a half-space of elastic perfectly-plastic metal, see Fig. 20. An approximate static analytical solution is given by Johnson in [12], assuming that contact pressure $p_m = 3\sigma_Y$ is constant and that the contact zone radius $r = \sqrt{(R\delta / 0.368)}$, δ being the penetration depth. The resulting approximated penetration force is thus inear with δ.

In the numerical simulation, only a cylinder of radius and height $3R$ is modelled instead of the full half space. The rigid indenter is represented by a material point with an associated parent pinball of radius R. This pinball is never splitted. One fourth of the cylinder (by symmetry) is discretized with 8192 cube elements and 360 prism elements (on the axis). Hierarchic pinballs of level 4 with CN and NC algorithms are associated with the relevant surface of the cylinder. A linear displacement in time is imposed to the penetrator, from 0 to R in 50 ms. Although this is not static loading, it has been verified that dynamic effects are negligible.

The computed penetration force agrees well with the approximate linear analytical solution, see Fig. 20b. The final shape of the solid is shown in Fig. 20c for the cylindrical 3D model, and in Fig. 20d for a cubes-only model, which is not exactly axisymmetric. Colors indicate the displacement norm. (More examples to be added here)

10 SUMMARY AND CONCLUSIONS

(To be written).

REFERENCES

1. Hallquist JO, Goudreau GL, Benson DJ. Sliding interfaces with contact-impact in large-scale Lagrangian computation. *Computer Methods in Applied Mechanics and Engineering* 1985; **51** : 107–137.
2. Benson DJ, Hallquist JO. A single surface contact algorithm for the postbuckling analysis of shell structures. Report to the University of California at San Diego, CA, 1987.
3. Belytschko T, Lin JI. A Three-Dimensional Impact-Penetration Algorithm with Erosion. *Computers and Structures* 1987 ; **25** : 95–104.
4. Belytschko T, Neal MO. Contact-Impact by the Pinball Algorithm with Penalty, Projection and Lagrangian Methods, in *Computational Techniques for Contact, Impact, Penetration and Perforation of Solids*, Winter Annual Meeting of the American Society of Mechanical Engineers, San Francisco, California, December 10-15, 1989 ; AMD **103**, 97–140.
5. Sarwas RE. Hidden Line Elimination and a Modified Pinball Algorithm for Finite Element Contact-Impact Problems with Shell Elements. Master's Thesis, Northwestern University, Evanston, Illinois, December 1989.
6. Belytschko T, Neal MO. Contact-Impact by the Pinball Algorithm with Penalty and Lagrangian Methods. *International Journal for Numerical Methods in Engineering* 1991; **31** : 547–572.
7. Belytschko T, Yeh IS. The Splitting Pinball Method for Contact-Impact Problems. *Computer Methods in Applied Mechanics and Engineering* 1993; **105** : 375–393.
8. Winkelmuller G. Auto-Contacts en Dynamique Rapide. In: Modélisation en Dynamique Rapide - De la Théorie à la Pratique, IPSI Course, Paris, 17-19 October 2000.
9. Lepareux M, Schwab B, Hoffmann A, Jamet P, Bung H. Un programme général pour l'analyse dynamique rapide—Cas des tuyauteries. Colloque: Tendances Actuelles en Calcul des Structures, Bastia, 6–8 November, 1985.
10. Casadei F. A Hierarchic Pinball Method for Contact-Impact in Fast Transient Dynamics, presented at the VI Congresso Nazionale della Società Italiana di Matematica Applicata e Industriale (SIMAI 2002), Chia (Cagliari), Italy, 27-31 May 2002.
11. Casadei F. A General Impact-Contact Algorithm Based on Hierarchic Pinballs for the EUROPLEXUS Software System. Joint Research Center Report N. I.03.176, December 2003.
12. Johnson KL. Contact Mechanics. Cambridge University Press, Chapter 6, pp. 153-201.

Table I. Uniform-step CD explicit time integration algorithm (Lagrangian description).

Algorithm UCD
0. Set initial conditions: $n \leftarrow 0$ (step counter), $t \leftarrow t^0$, $\mathbf{x} \leftarrow \mathbf{x}_0$, $\boldsymbol{\sigma} \leftarrow \boldsymbol{\sigma}_0$, $\mathbf{v} \leftarrow \mathbf{v}_0$, $W^{\text{int}} \leftarrow 0$, $W^{\text{ext}} \leftarrow W^{\text{kin}}$
1. GO TO 4.
2. End of the calculation.

Table III. Numerical solutions for the 1D step wave propagation.

Solution	Mesh/ Elements	Time step	Steps	Cycles	Max level frequency	Elements × cycles	CPU (s)
Step wave 1	Uniform/100	Uniform	250	—	—	2.5×10^4	0.2
Step wave 4	Graded/111	Partition	252	2,016	8	4.7×10^4	0.3

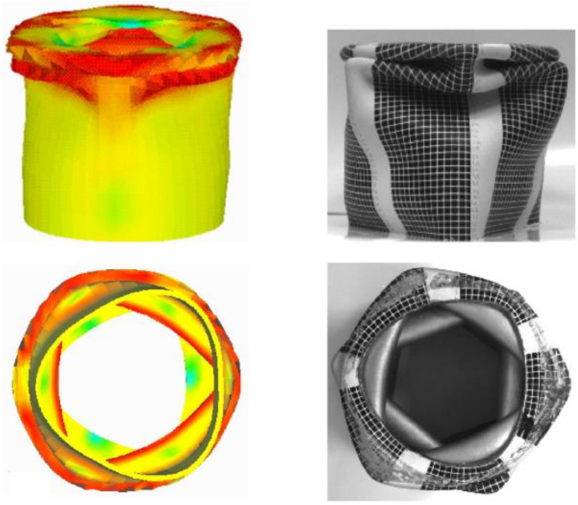


Figure 1. Crash of a metallic tube, simulation and experiment (Courtesy of CEA)

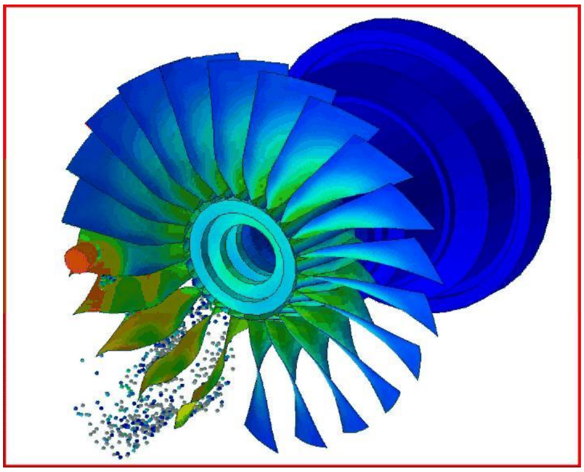
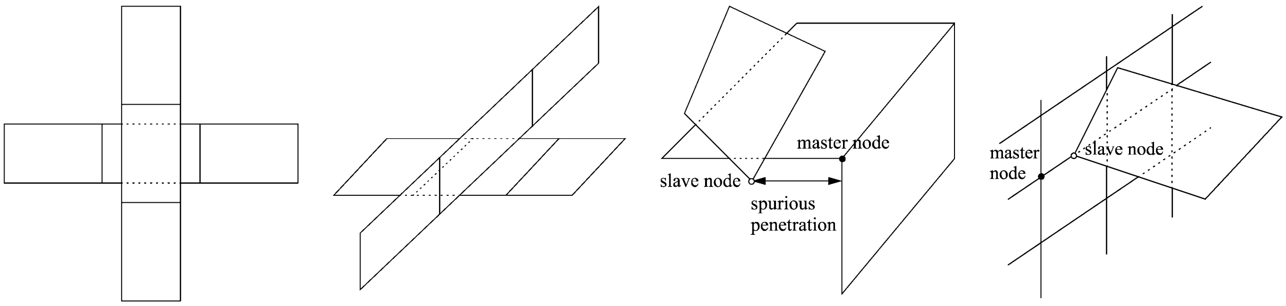


Figure 2. Simulation of bird strike on a jet fan by the SPH method (Courtesy of Snecma/CEA)



a) surface-to-surface contact b) edge-to-surface contact c) spurious penetration d) multiple master contacts

Figure 3. Examples of pathological contact detection by the sliding surface method (from [7])

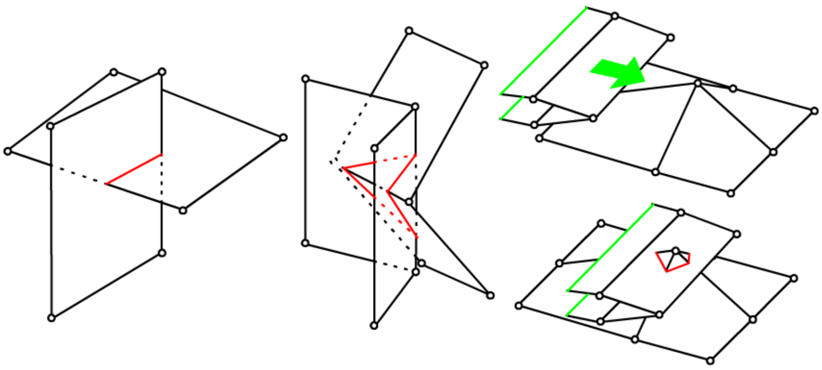
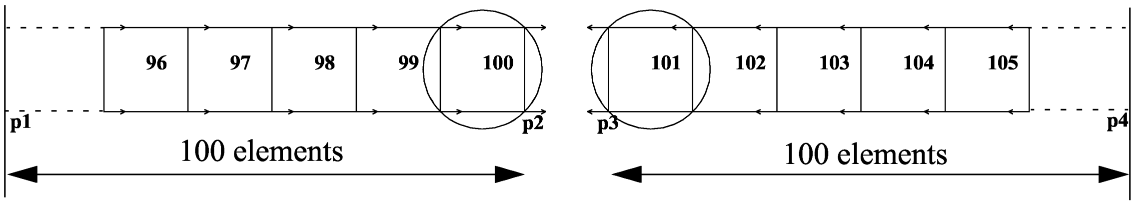
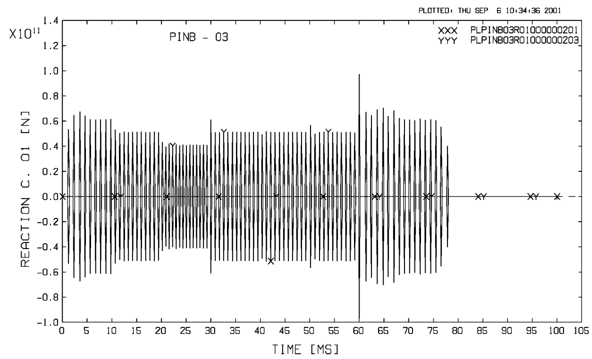


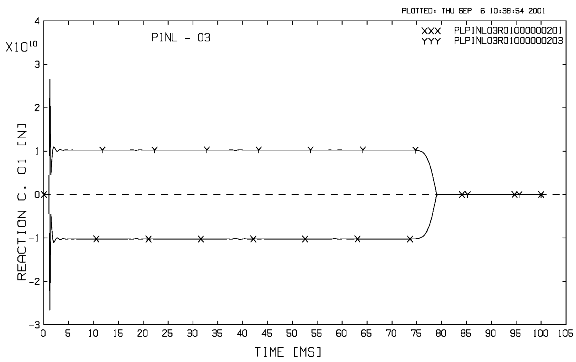
Figure 4. Further examples of pathological contact detection cases (from [8])



a) FE mesh with initial velocities and pinballs shown

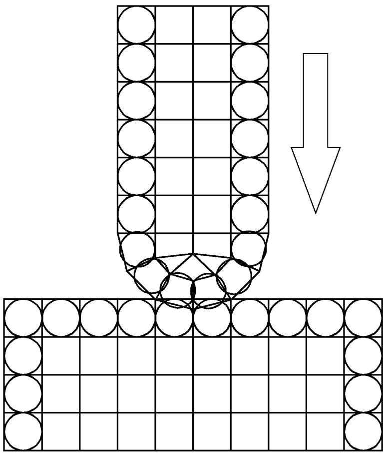


b) contact force obtained with the first algorithm

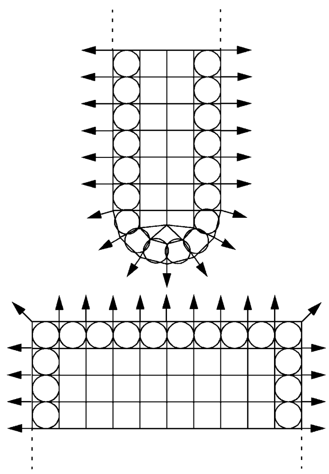


c) contact force obtained with the second algorithm

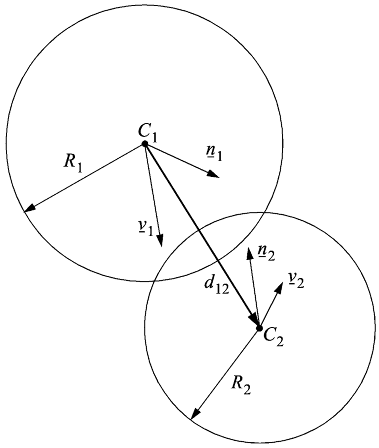
Figure 5. Bar impact test by the basic pinball method



a) the pinball concept shown in 2D

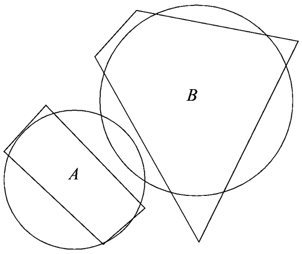


b) assembled surface normals

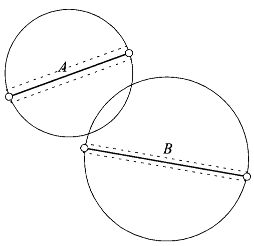


c) interpenetrating pinballs

Figure 6. The basic pinball method (from [6])



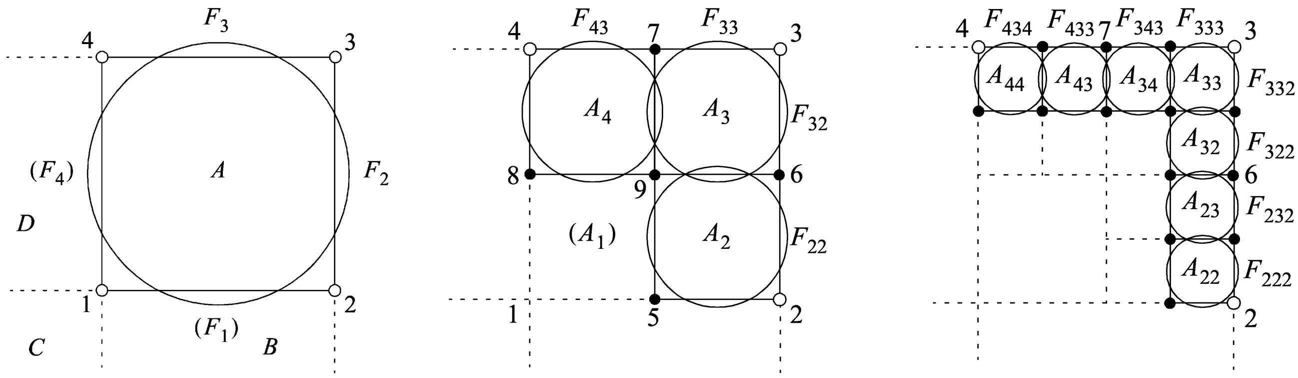
a) slender or distorted continuum elements



b) beam or shell elements

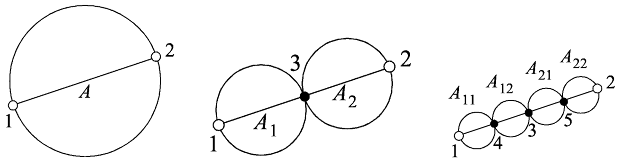
Figure 7. Shortcomings (incorrect contact detection) of the basic pinball method

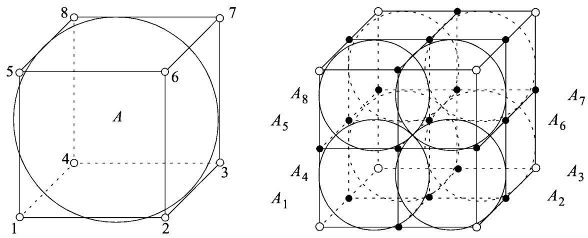
○ Mesh node ● Other geometric point



Level 0 | Level 1 | Level 2

a) Pinball splitting for a 2D quadrilateral



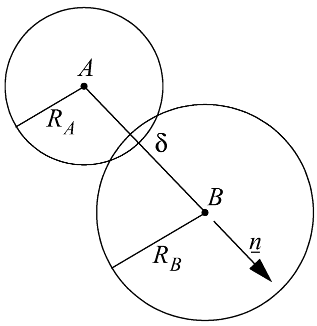


Level 0 | Level 1 | Level 2 | Level 0 | Level 1

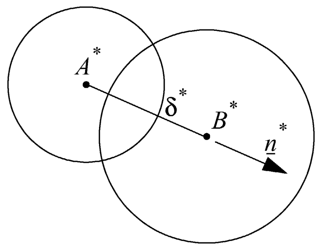
b) – Pinball splitting for a 2D beam or shell

c) – Pinball splitting for a 3D hexahedron

Figure 8. The hierarchic pinball method

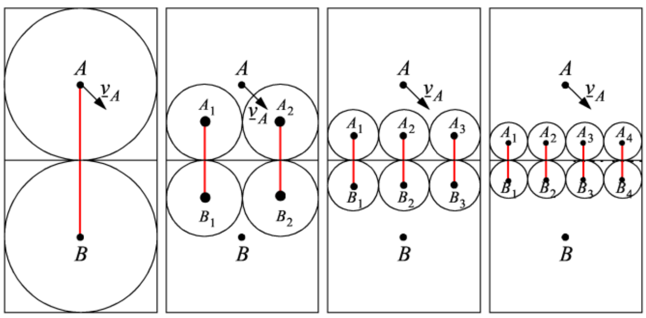


a) configuration at time station $n+1$



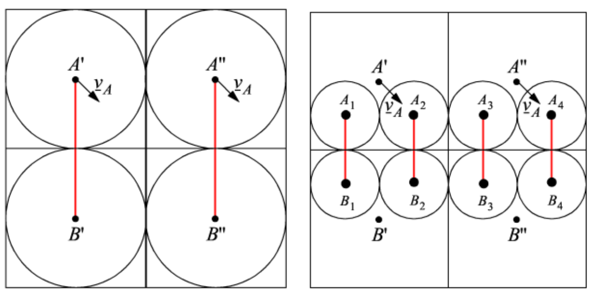
b) “free” configuration at time station $n+2$

Figure 9. *A priori* detection of rebound



Level 0 No redund. | Level 1 No redund. | (Level 1.5) 1 redund. | Level 2 2 redund.

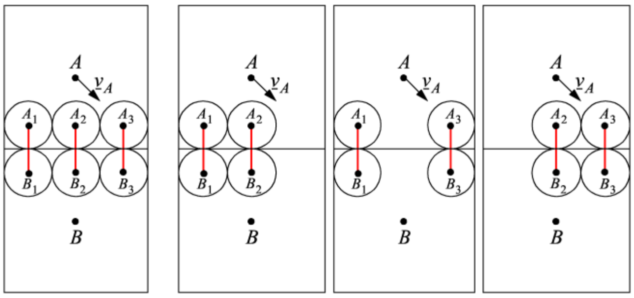
a) “flat” contact between a couple of elements



Level 0 No redundancies | Level 1 1 redundancy

b) “flat” contact between two element couples

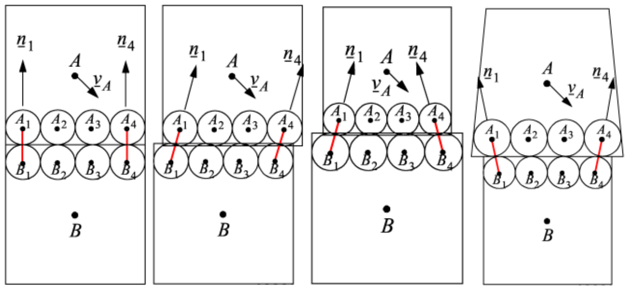
Figure 10. Example of redundant constraints generated by the hierarchic pinball method



(Level 1.5)
1 redun.

(Level 1.5)
No redundant constraints, equivalent results

Figure 11. Removing redundant constraints



Case 1
(ideal contact)

Case 2
(misalignment)

Case 3
(size mismatch)

Case 4
(deformation)

Figure 12. Examples of redundant constraints in flat contacts with hierarchic pinballs

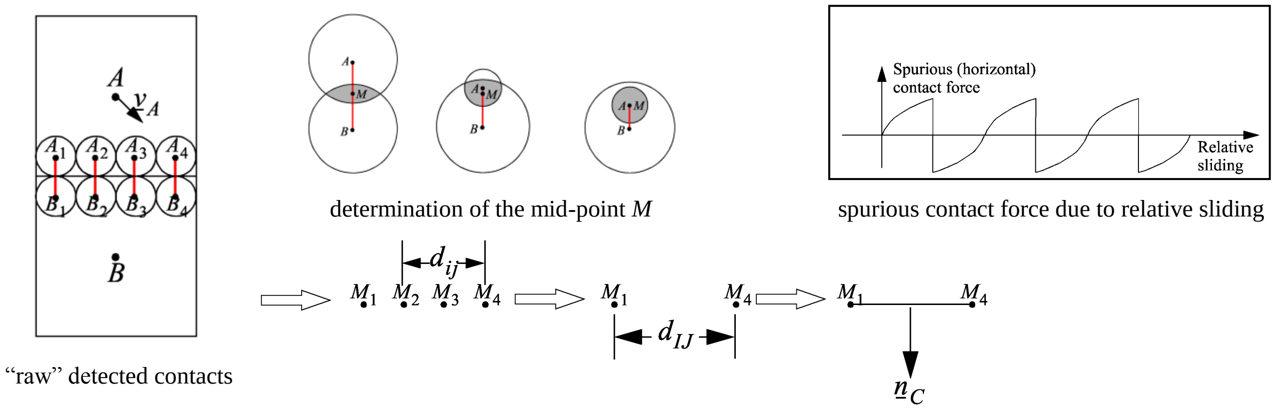
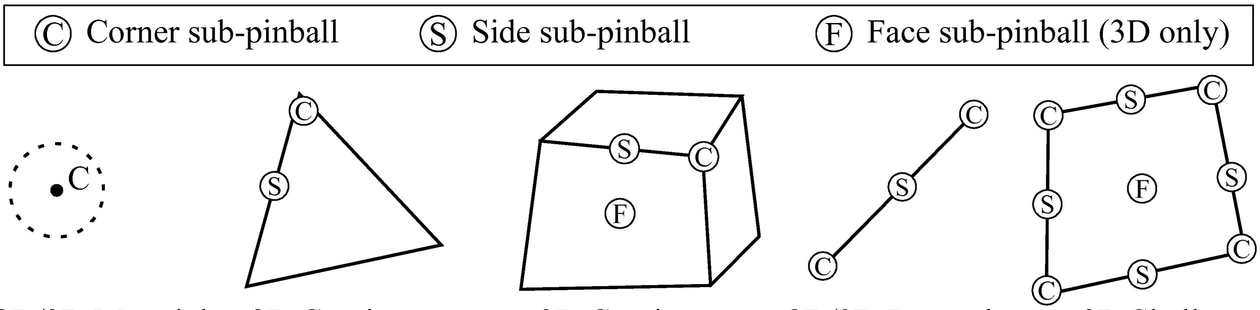


Figure 13. The common normal algorithm in a 2D case



2D/3D Material Points

2D Continuum Elements

3D Continuum Elements

2D/3D Bar and 2D Shell Elements

3D Shell Elements

Figure 14. Classification of descendent pinballs

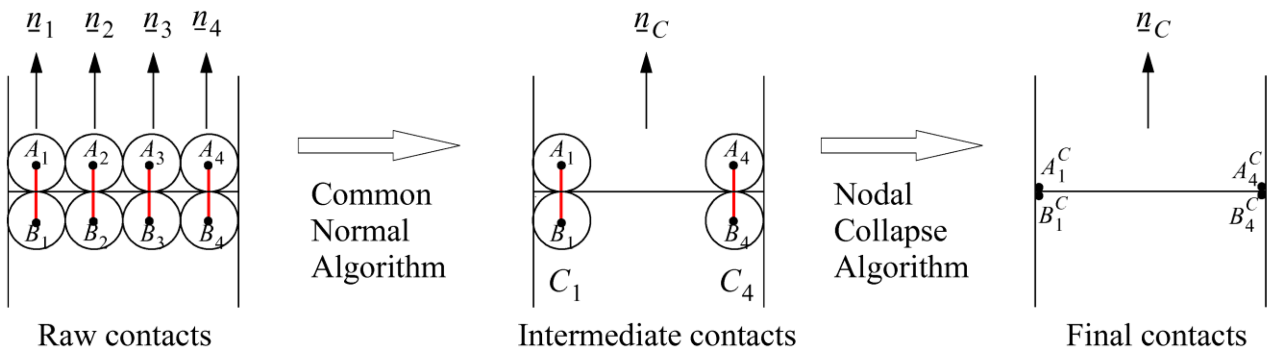


Figure 15. Nodal collapse of corner descendents in perfectly aligned flat contact

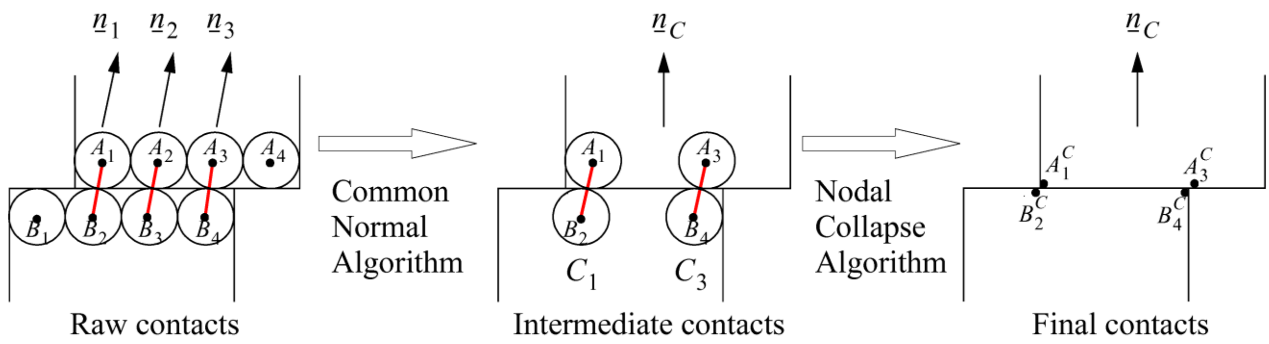
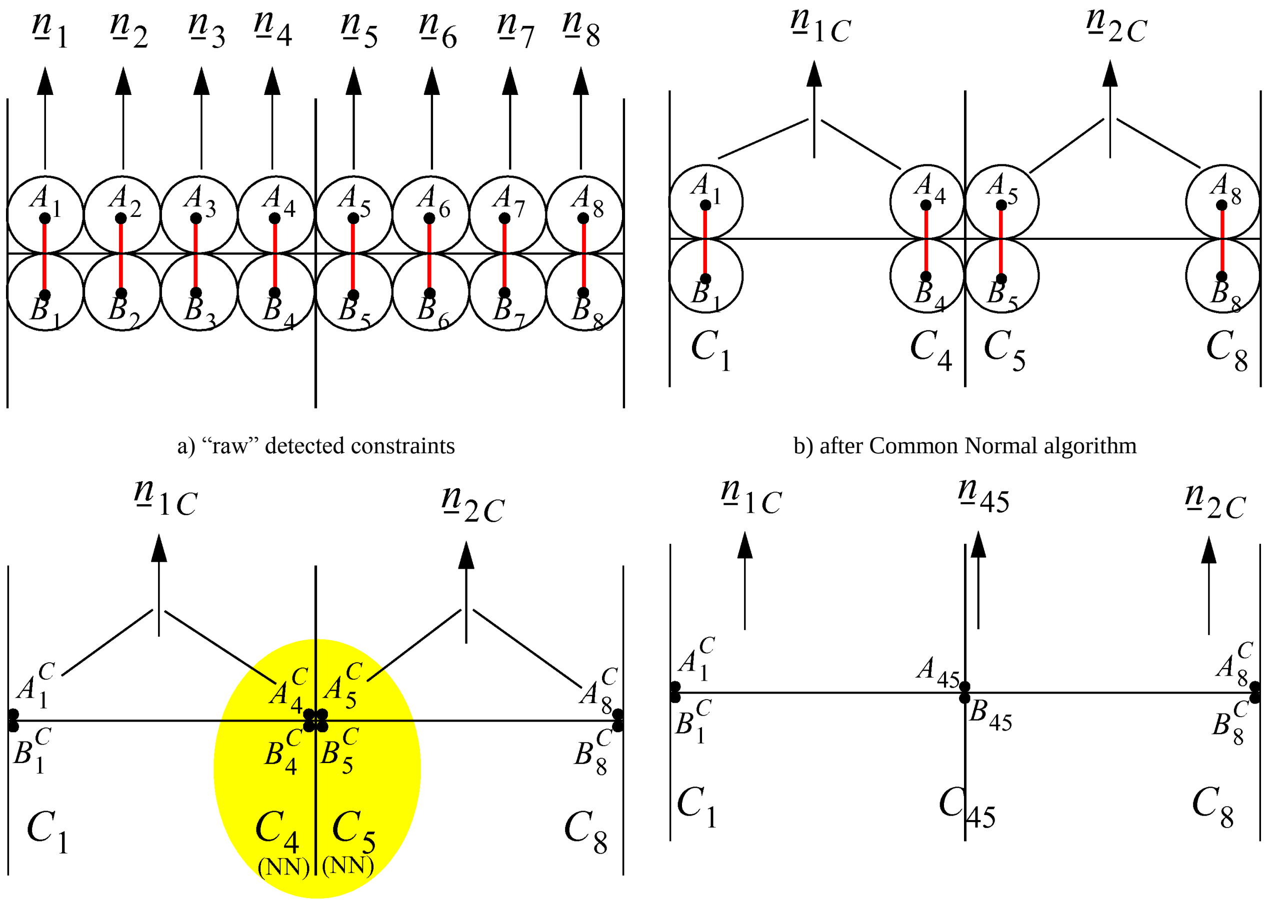


Figure 16. Nodal collapse of corner and side descendents in largely misaligned flat contact



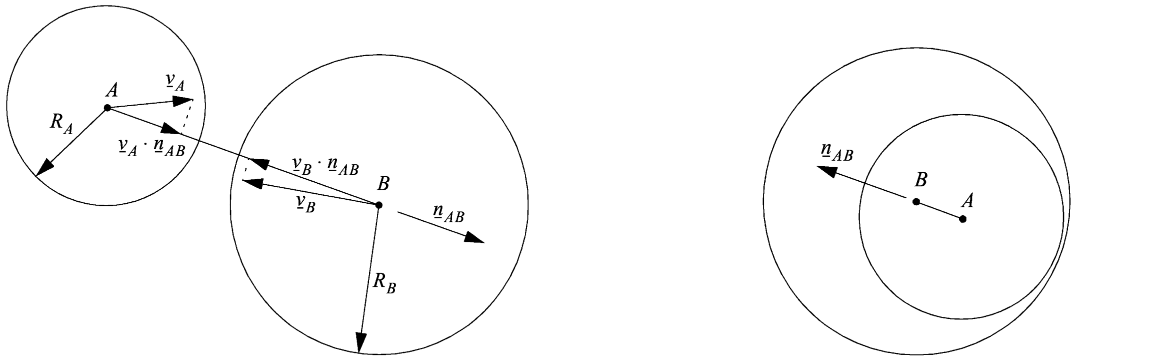
a) “raw” detected constraints

b) after Common Normal algorithm

c) after Nodal Collapse algorithm

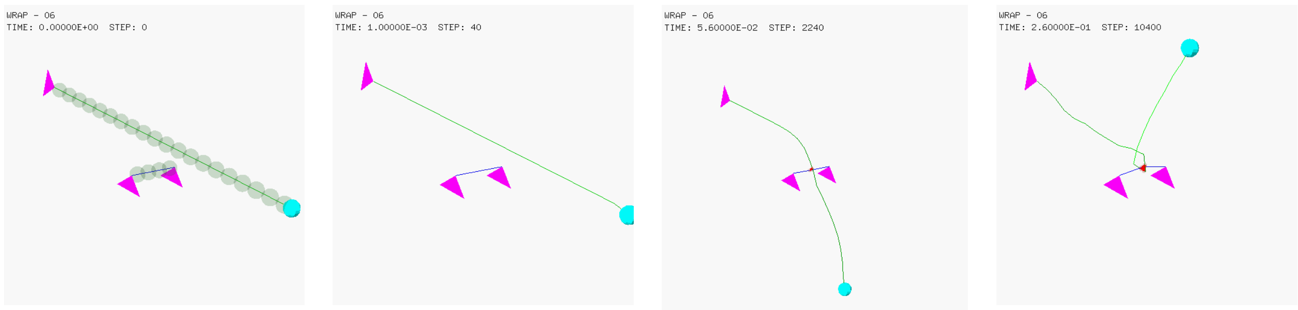
d) final constraints after redundancy elimination

Figure 17. Redundancy elimination in perfectly aligned multiple flat contact

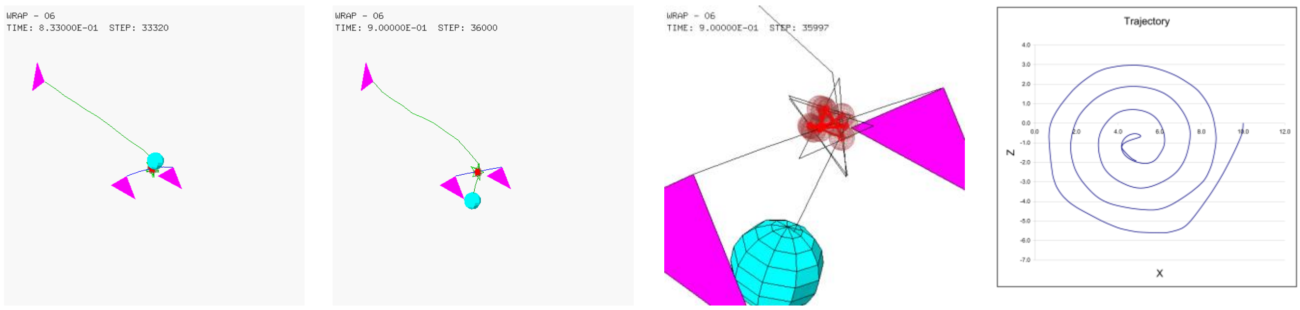


a) pinballs configuration at time station $n+1$ b) pathological configuration at time station $n+2$

Figure 18. Adaptation of the time step

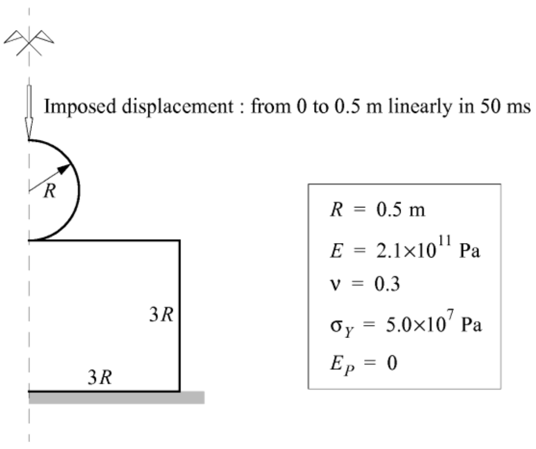


Geometry and parent pinballs | t = 1 ms | t = 56 ms (first contact) | t = 260 ms

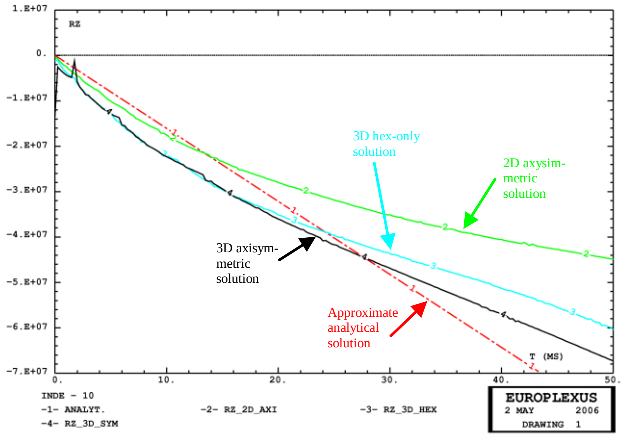


t = 833 ms (complete wrap) | t = 900 ms (rebound) | t = 900 ms (zoom) | Mass trajectory

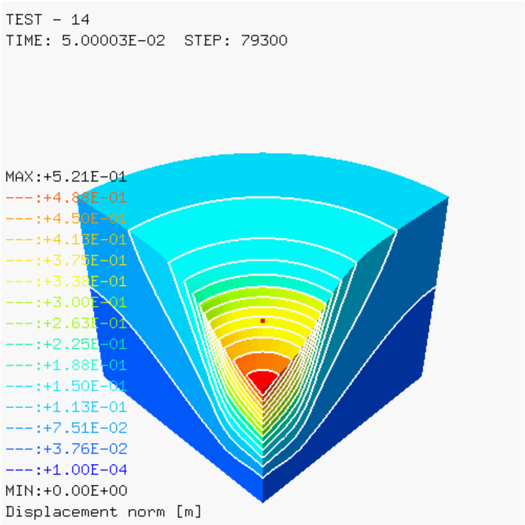
Figure 19. Cable wrapping example



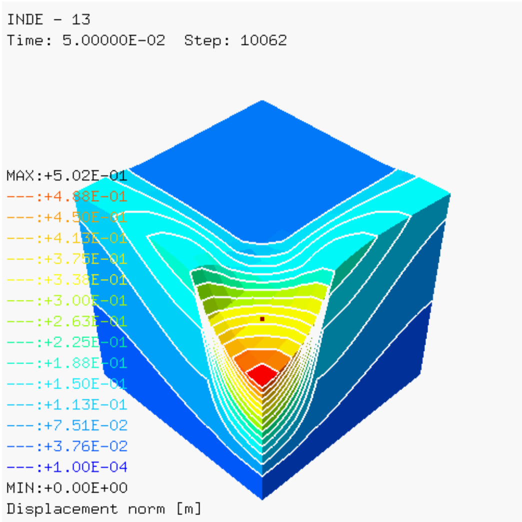
a) geometry and material



b) comparison of numerical and analytical solutions



c) final deformation



d) solution with all-hexahedral mesh

Figure 20. Sphere indentation example

9. Appendix C - Input Files

Sample input files

This Section contains, in alphabetical file order, the listings of all input files related to the examples which were proposed in the previous Sections.

asnc01.eco

```
STANDARD VERSION

 TITLE: ASNC01

      3> CONV win
      4>CPLA LAGR
      5>GEOM LIBR POIN 12 CAR1 4 TERM
 == SET FLAG IN INPUT FILE
 == GO BACK TO FLAG IN INPUT FILE
 == SET FLAG IN INPUT FILE
 == GO BACK TO FLAG IN INPUT FILE
      6> 0 0  1 0  2 0  0 1  1 1  2 1
      7> 0 1  1 1  2 1  0 2  1 2  2 2
      8> 1 2 5 4
      9> 2 3 6 5
     10> 7 8 11 10
     11> 8 9 12 11
     12>COMP COUL VERT LECT tous TERM
     13>MATE LINE RO 8000.0 YOUN 1.E12 NU 0.0
     14>          LECT tous TERM
     15>INIT VITE 2  50 LECT 1 PAS 1  6 TERM
     16>     VITE 2 -50 LECT 7 PAS 1 12 TERM
     17>OPTI PINS DUMP
     18>          STAT
     19>
     19>!VIDE
     20>
     20>!EQVL
     21>
     21>!EQVD
     22>          ASN
     23>
     23>!NORB
     24>LINK DECO PINB PENA SFAC 1.0
     25>               BODY LECT 1 2 TERM
     26>               BODY LECT 3 4 TERM
     27>ECRI COOR DEPL VITE ACCE FINT FEXT FLIA CONT ECRO FREQ 1
     28>     FICH ALIC FREQ 1
     29>OPTI NOTE
     30> CSTA 0.5E0
     31> LOG 1
     32>CALC TINI 0. TEND 100.E-3 NMAX 1
     33>*==============================================================
     34>PLAY

 -> END OF INITIALISATIONS ** TCPU =     0.02 SEC.

     35>CAME   1 EYE   1.00000E+00  1.00000E+00  9.23880E+00
     36>!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
     37>         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
     38>         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
     39>         UP    0.00000E+00  1.00000E+00  0.00000E+00
     40>         FOV   2.48819E+01
     41>!NAVIGATION MODE: ROTATING CAMERA
     42>!CENTER :  1.00000E+00  1.00000E+00  0.00000E+00
     43>!RSPHERE:  1.84776E+00
     44>!RADIUS :  9.23880E+00
     45>!ASPECT :  1.00000E+00
     46>!NEAR   :  7.20626E+00
     47>!FAR    :  1.29343E+01
     48>SCEN GEOM NAVI FREE
     49>          FACE HFRO
     50>          PINB PARE
     51>     COLO PAPE
     52>SLER CAM1 1 NFRA 1
     53>TRAC OFFS FICH BMP REND
     54>SCEN GEOM NAVI FREE
     55>          FACE HFRO
     56>          PINB NASN PASN NORM
     57>     COLO PAPE
     58>SLER CAM1 1 NFRA 1
     59>TRAC OFFS FICH BMP REND
     60>ENDPLAY
  >  END OF TIME STEP NO :       0 * T = 0.00000E+00 * TCPU =      0.03 SEC.
  >  END OF TIME STEP NO :       1 * T = 3.87298E-05 * TCPU =      0.05 SEC.
     61>*==============================================================
     62>FIN
   >  TOTAL CPU TIME :        0.05 SEC.
Saving image 0001 to asnc01_0001.bmp ... Checking OpenGL errors ...
bitmap saved.
Saving image 0002 to asnc01_0002.bmp ... Checking OpenGL errors ...
bitmap saved.
```

asnc01.epx

```
ASNC01
ECHO
 CONV win
CPLA LAGR
GEOM LIBR POIN 12 CAR1 4 TERM
 0 0  1 0  2 0  0 1  1 1  2 1
 0 1  1 1  2 1  0 2  1 2  2 2
 1 2 5 4
 2 3 6 5
 7 8 11 10
 8 9 12 11
COMP COUL VERT LECT tous TERM
MATE LINE RO 8000.0 YOUN 1.E12 NU 0.0
          LECT tous TERM
INIT VITE 2  50 LECT 1 PAS 1  6 TERM
     VITE 2 -50 LECT 7 PAS 1 12 TERM
OPTI PINS DUMP
          STAT
         !VIDE
         !EQVL
         !EQVD
          ASN
         !NORB
LINK DECO PINB PENA SFAC 1.0
               BODY LECT 1 2 TERM
               BODY LECT 3 4 TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA CONT ECRO FREQ 1
     FICH ALIC FREQ 1
OPTI NOTE
 CSTA 0.5E0
 LOG 1
CALC TINI 0. TEND 100.E-3 NMAX 1
*==============================================================
PLAY
CAME   1 EYE   1.00000E+00  1.00000E+00  9.23880E+00
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  1.00000E+00  1.00000E+00  0.00000E+00
!RSPHERE:  1.84776E+00
!RADIUS :  9.23880E+00
!ASPECT :  1.00000E+00
!NEAR   :  7.20626E+00
!FAR    :  1.29343E+01
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB CDES
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB NASN PASN NORM
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB DASN
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
ENDPLAY
*==============================================================
FIN
```

asnc02.epx

```
ASNC02
ECHO
 CONV win
CPLA LAGR
GEOM LIBR POIN 12 CAR1 4 TERM
 0 0  1 0  2 0  0 1  1 1  2 1
 0 1  1 1  2 1  0 2  1 2  2 2
 1 2 5 4
 2 3 6 5
 7 8 11 10
 8 9 12 11
COMP COUL VERT LECT tous TERM
MATE LINE RO 8000.0 YOUN 1.E12 NU 0.0
          LECT tous TERM
INIT VITE 2  50 LECT 1 PAS 1  6 TERM
     VITE 2 -50 LECT 7 PAS 1 12 TERM
OPTI PINS DUMP
          STAT
         !VIDE
         !EQVL
         !EQVD
          ASN
         !NORB
LINK DECO PINB PENA SFAC 1.0
               BODY MLEV 1 LECT 1 2 TERM
               BODY MLEV 1 LECT 3 4 TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA CONT ECRO FREQ 1
     FICH ALIC FREQ 1
OPTI NOTE
 CSTA 0.5E0
 LOG 1
CALC TINI 0. TEND 100.E-3 NMAX 1
*==============================================================
PLAY
CAME   1 EYE   1.00000E+00  1.00000E+00  9.23880E+00
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  1.00000E+00  1.00000E+00  0.00000E+00
!RSPHERE:  1.84776E+00
!RADIUS :  9.23880E+00
!ASPECT :  1.00000E+00
!NEAR   :  7.20626E+00
!FAR    :  1.29343E+01
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB CDES
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
```

```
          PINB NASN PASN NORM
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB DASN
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
ENDPLAY
*=================================================================
FIN
```

asnc03.epx

```
ASNC03
ECHO
 CONV win
CPLA LAGR
GEOM LIBR POIN 12 CAR1 4 TERM
 0 0  1 0  2 0  0 1  1 1  2 1
 0 1  1 1  2 1  0 2  1 2  2 2
 1 2 5 4
 2 3 6 5
 7 8 11 10
 8 9 12 11
COMP COUL VERT LECT tous TERM
MATE LINE RO 8000.0 YOUN 1.E12 NU 0.0
          LECT tous TERM
INIT VITE 2  50 LECT 1 PAS 1  6 TERM
     VITE 2 -50 LECT 7 PAS 1 12 TERM
OPTI PINS DUMP
          STAT
         !VIDE
         !EQVL
         !EQVD
          ASN
         !NORB
LINK DECO PINB PENA SFAC 1.0
               BODY MLEV 2 LECT 1 2 TERM
               BODY MLEV 2 LECT 3 4 TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA CONT ECRO FREQ 1
     FICH ALIC FREQ 1
OPTI NOTE
 CSTA 0.5E0
 LOG 1
CALC TINI 0. TEND 100.E-3 NMAX 1
*=================================================================
PLAY
CAME   1 EYE   1.00000E+00  1.00000E+00  9.23880E+00
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  1.00000E+00  1.00000E+00  0.00000E+00
!RSPHERE:  1.84776E+00
!RADIUS :  9.23880E+00
!ASPECT :  1.00000E+00
!NEAR   :  7.20626E+00
!FAR    :  1.29343E+01
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB CDES
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB NASN PASN NORM
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB DASN
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
ENDPLAY
*=================================================================
FIN
```

asnc04.eco

```
STANDARD VERSION

 TITLE: ASNC04

      3> CONV win
      4>CPLA LAGR
      5>GEOM LIBR POIN 18 CAR1 8 TERM
 == SET FLAG IN INPUT FILE
 == GO BACK TO FLAG IN INPUT FILE
 == SET FLAG IN INPUT FILE
 == GO BACK TO FLAG IN INPUT FILE
      6> 0 0  1 0  2 0
      7> 0 1  1 1  2 1
      8> 0 2  1 2  2 2
      9> 0 2  1 2  2 2
     10> 0 3  1 3  2 3
     11> 0 4  1 4  2 4
     12> 1 2 5 4
     13> 2 3 6 5
     14> 4 5 8 7
     15> 5 6 9 8
     16> 10 11 14 13
     17> 11 12 15 14
     18> 13 14 17 16
     19> 14 15 18 17
     20>COMP COUL VERT LECT tous TERM
     21>MATE LINE RO 8000.0 YOUN 1.E12 NU 0.0
     22>          LECT tous TERM
     23>INIT VITE 2  50 LECT  1 PAS 1  9 TERM
     24>     VITE 2 -50 LECT 10 PAS 1 18 TERM
     25>OPTI PINS DUMP
     26>          STAT
     27>
     27>!VIDE
     28>
     28>!EQVL
     29>
     29>!EQVD
     30>          ASN
     31>
     31>!NORB
     32>LINK DECO PINB PENA SFAC 1.0
     33>               BODY LECT 3 4 TERM
     34>               BODY LECT 5 6 TERM
     35>ECRI COOR DEPL VITE ACCE FINT FEXT FLIA CONT ECRO FREQ 1
     36>     FICH ALIC FREQ 1
     37>OPTI NOTE
     38> CSTA 0.5E0
     39> LOG 1
     40>CALC TINI 0. TEND 100.E-3 NMAX 1
     41>*==============================================================
     42>PLAY

 -> END OF INITIALISATIONS ** TCPU =     0.00 SEC.

     43>CAME   1 EYE   1.00000E+00  2.00000E+00  1.22036E+01
     44>!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
     45>         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
     46>         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
     47>         UP    0.00000E+00  1.00000E+00  0.00000E+00
     48>         FOV   2.48819E+01
     49>!NAVIGATION MODE: ROTATING CAMERA
     50>!CENTER :  1.00000E+00  2.00000E+00  0.00000E+00
     51>!RSPHERE:  2.44072E+00
     52>!RADIUS :  1.22036E+01
     53>!ASPECT :  1.00000E+00
     54>!NEAR   :  9.51880E+00
     55>!FAR    :  1.70850E+01
     56>SCEN GEOM NAVI FREE
     57>          FACE HFRO
     58>          PINB PARE
     59>     COLO PAPE
     60>SLER CAM1 1 NFRA 1
     61>TRAC OFFS FICH BMP REND
     62>SCEN GEOM NAVI FREE
     63>          FACE HFRO
     64>          PINB NASN PASN NORM
     65>     COLO PAPE
     66>SLER CAM1 1 NFRA 1
     67>TRAC OFFS FICH BMP REND
     68>ENDPLAY
  >  END OF TIME STEP NO :       0 * T = 0.00000E+00 * TCPU =      0.03 SEC.
  >  END OF TIME STEP NO :       1 * T = 3.87298E-05 * TCPU =      0.05 SEC.
     69>*==============================================================
     70>FIN
   >  TOTAL CPU TIME :        0.05 SEC.
Saving image 0001 to asnc04_0001.bmp ... Checking OpenGL errors ...
bitmap saved.
Saving image 0002 to asnc04_0002.bmp ... Checking OpenGL errors ...
bitmap saved.
```

asnc04.epx

```
ASNC04
ECHO
 CONV win
CPLA LAGR
GEOM LIBR POIN 18 CAR1 8 TERM
 0 0  1 0  2 0
 0 1  1 1  2 1
 0 2  1 2  2 2
 0 2  1 2  2 2
 0 3  1 3  2 3
 0 4  1 4  2 4
 1 2 5 4
 2 3 6 5
 4 5 8 7
 5 6 9 8
 10 11 14 13
 11 12 15 14
 13 14 17 16
 14 15 18 17
COMP COUL VERT LECT tous TERM
MATE LINE RO 8000.0 YOUN 1.E12 NU 0.0
          LECT tous TERM
INIT VITE 2  50 LECT  1 PAS 1  9 TERM
     VITE 2 -50 LECT 10 PAS 1 18 TERM
OPTI PINS DUMP
          STAT
         !VIDE
         !EQVL
         !EQVD
          ASN
         !NORB
LINK DECO PINB PENA SFAC 1.0
               BODY LECT 3 4 TERM
               BODY LECT 5 6 TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA CONT ECRO FREQ 1
     FICH ALIC FREQ 1
OPTI NOTE
 CSTA 0.5E0
 LOG 1
CALC TINI 0. TEND 100.E-3 NMAX 1
*=================================================================
PLAY
CAME   1 EYE   1.00000E+00  2.00000E+00  1.22036E+01
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
```

```
!CENTER :  1.00000E+00  2.00000E+00  0.00000E+00
!RSPHERE:  2.44072E+00
!RADIUS :  1.22036E+01
!ASPECT :  1.00000E+00
!NEAR   :  9.51880E+00
!FAR    :  1.70850E+01
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB CDES
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB NASN PASN NORM
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB DASN
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
ENDPLAY
*=================================================================
FIN
```

asnc05.epx

```
ASNC05
ECHO
 CONV win
CPLA LAGR
GEOM LIBR POIN 6 BARR 5 TERM
 0 0  1 1  2 1  3 1  4 2  4 0
 1 2
 2 3
 3 4
 4 5
 4 6
COMP EPAI 0.1 LECT tous TERM
     COUL VERT LECT tous TERM
MATE LINE RO 8000.0 YOUN 1.E12 NU 0.0
          LECT tous TERM
OPTI PINS DUMP
          STAT
         !VIDE
         !EQVL
         !EQVD
          ASN
         !NORB
LINK DECO PINB PENA SFAC 1.0
               BODY MLEV 0 LECT 1 PAS 1 5 TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA CONT ECRO FREQ 1
     FICH ALIC FREQ 1
OPTI NOTE
 CSTA 0.5E0
 LOG 1
CALC TINI 0. TEND 100.E-3 NMAX 1
*=================================================================
PLAY
CAME   1 EYE   2.00000E+00  1.00000E+00  1.30656E+01
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  2.00000E+00  1.00000E+00  0.00000E+00
!RSPHERE:  2.61313E+00
!RADIUS :  1.30656E+01
!ASPECT :  1.00000E+00
!NEAR   :  1.01912E+01
!FAR    :  1.82919E+01
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB PARE NASN PASN NORM
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB CDES DASN
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
ENDPLAY
*=================================================================
FIN
```

asnc06.epx

```
ASNC06
ECHO
 CONV win
CPLA LAGR
GEOM LIBR POIN 6 BARR 5 TERM
 0 0  1 1  2 1  3 1  4 2  4 0
 1 2
 2 3
 3 4
 4 5
 4 6
COMP EPAI 0.1 LECT tous TERM
     COUL VERT LECT tous TERM
MATE LINE RO 8000.0 YOUN 1.E12 NU 0.0
          LECT tous TERM
OPTI PINS DUMP
          STAT
          VIDE
         !EQVL
         !EQVD
          ASN
         !NORB
LINK DECO PINB PENA SFAC 1.0
               BODY MLEV 1 LECT 1 PAS 1 5 TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA CONT ECRO FREQ 1
     FICH ALIC FREQ 1
OPTI NOTE
 CSTA 0.5E0
 LOG 1
CALC TINI 0. TEND 100.E-3 NMAX 0
*=================================================================
PLAY
CAME   1 EYE   2.00000E+00  1.00000E+00  1.30656E+01
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  2.00000E+00  1.00000E+00  0.00000E+00
!RSPHERE:  2.61313E+00
!RADIUS :  1.30656E+01
!ASPECT :  1.00000E+00
!NEAR   :  1.01912E+01
!FAR    :  1.82919E+01
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB PARE NASN PASN
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB CDES DASN
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
ENDPLAY
*=================================================================
FIN
```

asnc07.epx

```
ASNC07
ECHO
 CONV win
TRID LAGR
GEOM LIBR POIN 12 Q4GS 5 TERM
 0 0 0   1 1 0   2 1 0   3 1 0   4 2 0   4 0 0
 0 0 1   1 1 1   2 1 1   3 1 1   4 2 1   4 0 1
 1 2 8 7
 2 3 9 8
 3 4 10 9
 4 5 11 10
 4 6 12 10
COMP EPAI 0.1 LECT tous TERM
     COUL VERT LECT tous TERM
MATE LINE RO 8000.0 YOUN 1.E12 NU 0.0
          LECT tous TERM
OPTI PINS DUMP
          STAT
         !VIDE
         !EQVL
         !EQVD
          ASN
         !NORB
LINK DECO PINB PENA SFAC 1.0
               BODY MLEV 0 LECT 1 PAS 1 5 TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA CONT ECRO FREQ 1
     FICH ALIC FREQ 1
OPTI NOTE
 CSTA 0.5E0
 LOG 1
CALC TINI 0. TEND 100.E-3 NMAX 0
*=================================================================
PLAY
CAME   1 EYE   1.03655E+00  1.00629E+01  1.15584E+01
!        Q     9.40165E-01 -3.33943E-01 -5.13715E-02 -4.39644E-02
         VIEW  6.72322E-02 -6.32439E-01 -7.71687E-01
         RIGH  9.90856E-01 -4.83573E-02  1.25959E-01
         UP    1.16978E-01  7.73099E-01 -6.23405E-01
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  2.00000E+00  1.00000E+00  5.00000E-01
!RSPHERE:  2.86603E+00
!RADIUS :  1.43301E+01
!ASPECT :  1.00000E+00
!NEAR   :  1.11775E+01
!FAR    :  2.00622E+01
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB PARE NASN PASN
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB CDES DASN
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
ENDPLAY
*=================================================================
FIN
```

asnc08.epx

```
ASNC08
ECHO
 CONV win
TRID LAGR
GEOM LIBR POIN 12 Q4GS 5 TERM
 0 0 0   1 1 0   2 1 0   3 1 0   4 2 0   4 0 0
```

```
 0 0 1   1 1 1   2 1 1   3 1 1   4 2 1   4 0 1
 1 2 8 7
 2 3 9 8
 3 4 10 9
 4 5 11 10
 4 6 12 10
COMP EPAI 0.1 LECT tous TERM
     COUL VERT LECT tous TERM
MATE LINE RO 8000.0 YOUN 1.E12 NU 0.0
          LECT tous TERM
OPTI PINS DUMP
          STAT
          VIDE
         !EQVL
         !EQVD
          ASN
         !NORB
LINK DECO PINB PENA SFAC 1.0
               BODY MLEV 1 LECT 1 PAS 1 5 TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA CONT ECRO FREQ 1
     FICH ALIC FREQ 1
OPTI NOTE
 CSTA 0.5E0
 LOG 1
CALC TINI 0. TEND 100.E-3 NMAX 1
*===================================================================
PLAY
CAME   1 EYE   1.03655E+00  1.00629E+01  1.15584E+01
!        Q      9.40165E-01 -3.33943E-01 -5.13715E-02 -4.39644E-02
         VIEW  6.72322E-02 -6.32439E-01 -7.71687E-01
         RIGH  9.90856E-01 -4.83573E-02  1.25959E-01
         UP    1.16978E-01  7.73099E-01 -6.23405E-01
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  2.00000E+00  1.00000E+00  5.00000E-01
!RSPHERE:  2.86603E+00
!RADIUS :  1.43301E+01
!ASPECT :  1.00000E+00
!NEAR   :  1.11775E+01
!FAR    :  2.00622E+01
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB PARE NASN PASN
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB CDES DASN
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
ENDPLAY
*===================================================================
FIN
```

asnc09.epx

```
ASNC09
ECHO
 CONV win
TRID LAGR
GEOM LIBR POIN 12 Q4GS 5 TERM
 0 0 0   1 1 0   2 1 0   3 1 0   4 2 0   4 0 0
 0 0 1   1 1 1   2 1 1   3 1 1   4 2 1   4 0 1
 1 2 8 7
 2 3 9 8
 3 4 10 9
 4 5 11 10
 4 6 12 10
COMP EPAI 0.1 LECT tous TERM
     COUL VERT LECT tous TERM
MATE LINE RO 8000.0 YOUN 1.E12 NU 0.0
          LECT tous TERM
OPTI PINS DUMP
          STAT
          VIDE
         !EQVL
         !EQVD
          ASN
         !NORB
LINK DECO PINB PENA SFAC 1.0
               BODY MLEV 2 LECT 1 PAS 1 5 TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA CONT ECRO FREQ 1
     FICH ALIC FREQ 1
OPTI NOTE
 CSTA 0.5E0
 LOG 1
CALC TINI 0. TEND 100.E-3 NMAX 1
*===================================================================
PLAY
CAME   1 EYE   1.03655E+00  1.00629E+01  1.15584E+01
!        Q      9.40165E-01 -3.33943E-01 -5.13715E-02 -4.39644E-02
         VIEW  6.72322E-02 -6.32439E-01 -7.71687E-01
         RIGH  9.90856E-01 -4.83573E-02  1.25959E-01
         UP    1.16978E-01  7.73099E-01 -6.23405E-01
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  2.00000E+00  1.00000E+00  5.00000E-01
!RSPHERE:  2.86603E+00
!RADIUS :  1.43301E+01
!ASPECT :  1.00000E+00
!NEAR   :  1.11775E+01
!FAR    :  2.00622E+01
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB PARE NASN PASN
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB CDES DASN
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
ENDPLAY
*===================================================================
FIN
```

asnc10.epx

```
ASNC10
ECHO
 CONV win
TRID LAGR
GEOM LIBR POIN 12 Q4GS 5 TERM
 0 0 0   1 1 0   2 1 0   3 1 0   4 2 0   4 0 0
 0 0 1   1 1 1   2 1 1   3 1 1   4 2 1   4 0 1
 1 2 8 7
 2 3 9 8
 3 4 10 9
 4 5 11 10
 4 6 12 10
COMP EPAI 0.1 LECT tous TERM
     COUL VERT LECT tous TERM
MATE LINE RO 8000.0 YOUN 1.E12 NU 0.0
          LECT tous TERM
OPTI PINS DUMP
          STAT
          VIDE
         !EQVL
         !EQVD
          ASN
         !NORB
LINK DECO PINB PENA SFAC 1.0
               BODY MLEV 3 LECT 1 PAS 1 5 TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA CONT ECRO FREQ 1
     FICH ALIC FREQ 1
OPTI NOTE
 CSTA 0.5E0
 LOG 1
CALC TINI 0. TEND 100.E-3 NMAX 1
*===================================================================
PLAY
CAME   1 EYE   1.03655E+00  1.00629E+01  1.15584E+01
!        Q      9.40165E-01 -3.33943E-01 -5.13715E-02 -4.39644E-02
         VIEW  6.72322E-02 -6.32439E-01 -7.71687E-01
         RIGH  9.90856E-01 -4.83573E-02  1.25959E-01
         UP    1.16978E-01  7.73099E-01 -6.23405E-01
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  2.00000E+00  1.00000E+00  5.00000E-01
!RSPHERE:  2.86603E+00
!RADIUS :  1.43301E+01
!ASPECT :  1.00000E+00
!NEAR   :  1.11775E+01
!FAR    :  2.00622E+01
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB PARE NASN PASN
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB CDES DASN
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
ENDPLAY
*===================================================================
FIN
```

asno00.epx

```
ASNO00
ECHO
!CONV win
CPLA LAGR
GEOM LIBR POIN 9 CAR1 3 TRIA 2 TERM
 .0 .0  .1 .0
 .0 .1  .1 .1  .2 .1  .3 .1
 .1 .2  .2 .2  .3 .2
 1 2 4 3
 4 5 8 7
 5 6 9 8
 2 5 4
 3 4 7
MATE LINE RO 8000.0 YOUN 1.E12 NU 0.0
          LECT tous TERM
opti dump
OPTI PINS dump
          stat
         !vide
         !EQVL
          EQVD
LINK COUP PINB BODY LECT tous TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA CONT ECRO FREQ 1
     FICH ALIC TEMP FREQ 1
                    POIN LECT tous TERM
                    ELEM LECT tous TERM
OPTI NOTE
 LOG 1
CALCUL TINI 0. TEND 100.E-3 NMAX 15
FIN
```

asno01.epx

```
ASNO01
ECHO
 CONV win
CPLA LAGR
GEOM LIBR POIN 9 CAR1 3 TRIA 2 TERM
 .0 .0  .1 .0
 .0 .1  .1 .1  .2 .1  .3 .1
 .1 .2  .2 .2  .3 .2
```

```
 1 2 4 3
 4 5 8 7
 5 6 9 8
 2 5 4
 3 4 7
MATE LINE RO 8000.0 YOUN 1.E12 NU 0.0
          LECT tous TERM
opti dump
OPTI PINS dump
          stat
         !vide
         !EQVL
          EQVD
          ASN
LINK COUP PINB BODY LECT tous TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA CONT ECRO FREQ 1
     FICH ALIC TEMP FREQ 1
                    POIN LECT tous TERM
                    ELEM LECT tous TERM
OPTI NOTE
 LOG 1
CALCUL TINI 0. TEND 100.E-3 NMAX 15
*=================================================================
PLAY
CAME   1 EYE   1.50000E-01  1.00000E-01  1.10983E+00
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
CAME   2 EYE   1.39754E-01  1.06421E-01  7.93279E-01
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
SCEN GEOM NAVI FREE
          FACE HFRO
     PINB NASN PASN
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP OBJE LECT tous TERM REND
SCEN GEOM NAVI FREE
          FACE HFRO
     PINB NASN PASN
     COLO PAPE
SLER CAM1 2 NFRA 1
TRAC OFFS FICH BMP OBJE LECT 2 4 TERM REND
SCEN GEOM NAVI FREE
          FACE HFRO
     PINB PARE NASN PASN
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP OBJE LECT tous TERM REND
SCEN GEOM NAVI FREE
          FACE HFRO
     PINB PARE NASN PASN
     COLO PAPE
SLER CAM1 2 NFRA 1
TRAC OFFS FICH BMP OBJE LECT 2 4 TERM REND
ENDPLAY
*=================================================================
FIN
```

asno02.epx

```
ASNO02
ECHO
!CONV win
TRID LAGR
GEOM LIBR POIN 24 CUB8 4 TERM
 0 0 0  1 0 0
 0 1 0  1 1 0
 0 0 1  1 0 1
 0 1 1  1 1 1
 0 0 2  1 0 2
 0 1 2  1 1 2
 0 0 2.07  1 0 2.07
 0 1 2.07  1 1 2.07
 0 0 3.07  1 0 3.07
 0 1 3.07  1 1 3.07
 0 0 4.07  1 0 4.07
 0 1 4.07  1 1 4.07
  1  2  4  3  5  6  8  7
  5  6  8  7  9 10 12 11
 13 14 16 15 17 18 20 19
 17 18 20 19 21 22 24 23
MATE VM23 RO 8000.0 YOUN 1.E12 NU 0.0 ELAS 1.E12
          TRAC 1 1.E12 1.D0
          LECT tous TERM
INIT VITE 3  50 LECT  1 PAS 1 12 TERM
     VITE 3 -50 LECT 13 PAS 1 24 TERM
opti dump
OPTI PINS dump
          stat
         !vide
         !EQVL
          EQVD
          ASN
         !NORB
LINK DECO PINB PENA SFAC 1.0
               BODY MLEV 2 LECT 2 TERM
               BODY MLEV 2 LECT 3 TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA CONT ECRO FREQ 1
     FICH ALIC TEMP FREQ 1
                    POIN LECT tous TERM
                    ELEM LECT tous TERM
OPTI NOTE
 LOG 1
CALCUL TINI 0. TEND 100.E-3 NMAX 15
*=================================================================
PLAY
CAME   1 EYE  -9.99789E+00  3.75894E+00  6.52991E+00
!        Q     5.23877E-01  3.08719E-01 -4.64158E-01 -6.44052E-01
         VIEW  8.83986E-01 -2.74422E-01 -3.78499E-01
         RIGH -2.60491E-01 -9.61397E-01  8.86612E-02
         UP    3.88218E-01 -2.02203E-02  9.21346E-01
         FOV   2.48819E+01
CAME   2 EYE  -9.99789E+00  3.75894E+00  6.52991E+00
!        Q     5.23877E-01  3.08719E-01 -4.64158E-01 -6.44052E-01
         VIEW  8.83986E-01 -2.74422E-01 -3.78499E-01
         RIGH -2.60491E-01 -9.61397E-01  8.86612E-02
         UP    3.88218E-01 -2.02203E-02  9.21346E-01
         FOV   0.80000E+01
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB PARE NASN PASN
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
FREQ 6
GO
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB CDES DASN
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB DASN
     COLO PAPE
SLER CAM1 2 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB NORM DASN
     COLO PAPE
SLER CAM1 2 NFRA 1
TRAC OFFS FICH BMP REND
FREQ 15
GO
ENDPLAY
*=================================================================
SUIT
Post-treatment (time curves from alice temps file)
ECHO
*
RESU ALIC TEMP GARD PSCR
*
SORT GRAP
*
AXTE 1.0 'Time [s]'
*
COUR  1 'dz_9'  DEPL COMP 3 NOEU LECT  9 TERM
COUR  2 'dz_10' DEPL COMP 3 NOEU LECT 10 TERM
COUR  3 'dz_13' DEPL COMP 3 NOEU LECT 13 TERM
COUR  4 'dz_14' DEPL COMP 3 NOEU LECT 14 TERM
COUR 11 'fz_9'  FEXT COMP 3 NOEU LECT  9 TERM
COUR 12 'fz_10' FEXT COMP 3 NOEU LECT 10 TERM
COUR 13 'fz_13' FEXT COMP 3 NOEU LECT 13 TERM
COUR 14 'fz_14' FEXT COMP 3 NOEU LECT 14 TERM
*
trac 1 2 axes 1.0 'DISPL. [M]'
trac 3 4 axes 1.0 'DISPL. [M]'
trac 1 2 3 4 axes 1.0 'DISPL. [M]'
trac 11 12 axes 1.0 'FEXT [N]'
trac 13 14 axes 1.0 'FEXT [N]'
trac 11 12 13 14 axes 1.0 'FEXT [N]'
*=================================================================
FIN
```

asno03.dgibi

```
opti echo 1 dime 2 elem qua4;
opti trac psc ftra 'asno03_mesh.ps';
opti sauv form 'asno03.msh';
p0 = 0 0;
p1 = 1 0;
p2 = 1 0.5;
p3 = 0 0.5;
c1 = p0 d 20 p1;
c2 = p1 d 10 p2;
c3 = p2 d 20 p3;
c4 = p3 d 10 p0;
stru1 = dall c1 c2 c3 c4 plan;
stru2 = stru1 plus (0 0.6);
con1 = cont stru1;
con2 = cont stru2;
econ1 = stru1 elem appu larg con1;
econ2 = stru2 elem appu larg con2;
mesh = stru1 et stru2;
tass mesh;
sauv form mesh;
trac qual mesh;
trac qual (econ1 et econ2);
fin;
```

asno03.epx

```
ASNO03
ECHO
!CONV win
CAST mesh
CPLA LAGR
GEOM CAR1 stru1 stru2 TERM
MATE LINE RO 8000.0 YOUN 1.E12 NU 0.0
          LECT tous TERM
OPTI PINS EQVD ASN
LINK COUP PINB BODY LECT econ1 TERM
               BODY LECT econ2 TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA CONT ECRO FREQ 1
     FICH ALIC TEMP FREQ 1
                    POIN LECT tous TERM
                    ELEM LECT tous TERM
OPTI NOTE
CSTA 0.5E0
 LOG 1
CALCUL TINI 0. TEND 100.E-3 NMAX 15
```

```
*=================================================================
PLAY
CAME   1 EYE   5.00000E-01  5.50000E-01  3.79378E+00
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB PARE
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB NASN PASN
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
ENDPLAY
*=================================================================
FIN
```

asno04.dgibi

```
opti echo 1 dime 2 elem qua4;
opti trac psc ftra 'asno04_mesh.ps';
opti sauv form 'asno04.msh';
p0 = 0 0;
p1 = 1 0;
p2 = 2 0;
tol = 1.E-4;
c1 = p1 d 6 p2;
stru1 = c1 rota 30 180.0 p0;
stru2 = stru1 tour 180.0 p0;
stru = stru1 et stru2;
elim tol stru;
con = cont stru;
econ = stru elem appu larg con;
mesh = stru;
tass mesh;
sauv form mesh;
trac qual mesh;
trac qual econ;
fin;
```

asno04.epx

```
ASNO04
ECHO
!CONV win
CAST mesh
CPLA LAGR
GEOM CAR1 stru TERM
MATE LINE RO 8000.0 YOUN 1.E12 NU 0.0
          LECT tous TERM
OPTI PINS EQVD ASN
LINK COUP PINB BODY LECT econ TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA CONT ECRO FREQ 1
     FICH ALIC TEMP FREQ 1
                    POIN LECT tous TERM
                    ELEM LECT tous TERM
OPTI NOTE
CSTA 0.5E0
 LOG 1
CALCUL TINI 0. TEND 100.E-3 NMAX 15
*=================================================================
PLAY
CAME   1 EYE   0.00000E+00  0.00000E+00  1.44768E+01
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB PARE
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB NASN PASN
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
ENDPLAY
*=================================================================
FIN
```

asno05.epx

```
ASNO05
ECHO
 CONV win
CPLA LAGR
GEOM LIBR POIN 2 ED01 1 TERM
  0 0  1 1
  1 2
COMP EPAI 0.1  LECT tous TERM
     COUL VERT LECT tous TERM
MATE FANT 8000.0 LECT tous TERM
OPTI PINS dump
          stat
          vide
         !EQVL
         !EQVD
          ASN
LINK COUP PINB BODY LECT tous TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA CONT ECRO FREQ 1
     FICH ALIC TEMP FREQ 1
                    POIN LECT tous TERM
                    ELEM LECT tous TERM
OPTI NOTE
 LOG 1
CALC TINI 0. TEND 100.E-3
FIN
```

bar201.dgibi

```
opti echo 1;
opti dime 2 elem qua4;
opti sauv form 'bar201.msh';
opti trac psc ftra 'bar201_mesh.ps';
p0 = 0 0;
p1 = 0 0.01;
p2 = 1 0;
p3 = 1 0.01;
p4 = 1.0045 0;
p5 = 1.0045 0.01;
p6 = 2.0045 0;
p7 = 2.0045 0.01;
n1 = 1;
n2 = 100;
tol = 1.E-5;
c1 = p0 d n1 p1;
bar1 = c1 tran n2 p2;
bar2 = bar1 plus (1.0045 0);
bars = bar1 et bar2;
elim tol (bars et p0 et p1 et p2 et p3 et p4 et p5 et p6 et p7);
con1 = bar1 elem cont p2;
con2 = bar2 elem cont p4;
mesh = bars;
tass mesh;
sauv form mesh;
trac qual mesh;
trac qual (con1 et con2);
fin;
```

bar201.epx

```
BAR201
ECHO
!CONV win
CAST mesh
CPLA LAGR
GEOM CAR1 mesh TERM
COMP COUL VERT LECT tous TERM
MATE LINE RO 8000.0 YOUN 1.E12 NU 0.0
          LECT tous TERM
OPTI PINS STAT
         !ASN
LINK COUP PINB BODY LECT con1 TERM
               BODY LECT con2 TERM
INIT VITE 1  50 LECT bar1 TERM
     VITE 1 -50 LECT bar2 TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA CONT ECRO TFRE 0.4E-3
     FICH ALIC FREQ 1
OPTI NOTE
 CSTA 0.5E0
 LOG 1
CALC TINI 0. TEND 0.4E-3
*=================================================================
!PLAY
!CAME   1 EYE   5.00000E-01  2.21000E+00  1.21286E+01
!!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
!         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
!         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
!         UP    0.00000E+00  1.00000E+00  0.00000E+00
!         FOV   2.48819E+01
!FREQ 2
!GO
!SCEN GEOM NAVI FREE
!          FACE HFRO
!          PINB PARE
!     COLO PAPE
!SLER CAM1 1 NFRA 1
!TRAC OFFS FICH BMP REND
!SCEN GEOM NAVI FREE
!          FACE HFRO
!          PINB NASN PASN
!     COLO PAPE
!SLER CAM1 1 NFRA 1
!TRAC OFFS FICH BMP REND
!SCEN GEOM NAVI FREE
!          FACE HFRO
!          PINB NORM
!     COLO PAPE
!SLER CAM1 1 NFRA 1
!TRAC OFFS FICH BMP REND
!FREQ 30
!GO
!ENDPLAY
*=================================================================
SUIT
Post-treatment (time curves from alice file)
ECHO
RESU ALIC GARD PSCR
SORT GRAP
AXTE 1.0 'Time [s]'
COUR  1 'dx_p0' DEPL COMP 1 NOEU LECT p0 TERM
COUR  2 'dx_p2' DEPL COMP 1 NOEU LECT p2 TERM
COUR  3 'dx_p4' DEPL COMP 1 NOEU LECT p4 TERM
COUR  4 'dx_p6' DEPL COMP 1 NOEU LECT p6 TERM
COUR  5 'vx_p0' VITE COMP 1 NOEU LECT p0 TERM
COUR  6 'vx_p2' VITE COMP 1 NOEU LECT p2 TERM
COUR  7 'vx_p4' VITE COMP 1 NOEU LECT p4 TERM
COUR  8 'vx_p6' VITE COMP 1 NOEU LECT p6 TERM
TRAC 1 2 3 4 AXES 1.0 'DISPL. [M]' YZER
TRAC 5 6 7 8 AXES 1.0 'VELOC. [M/S]' YZER
*=================================================================
FIN
```

bar201a.epx

```
BAR201 Post-treatment (animation from alice file)
ECHO
RESU ALIC 'bar201.ali' GARD PSCR
SORT VISU NSTO 1
*=================================================================
PLAY
CAME   1 EYE   3.00000E+00  3.20711E+00  2.90257E+01
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
SCEN GEOM NAVI FREE
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH AVI NOCL NFTO 1001 FPS 25 KFRE 10 COMP -1 REND
FREQ 1
GOTR LOOP 999 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDPLAY
*=================================================================
FIN
```

bar202.dgibi

```
opti echo 1;
opti dime 2 elem qua4;
opti sauv form 'bar202.msh';
opti trac psc ftra 'bar202_mesh.ps';
p0 = 0 0;
p1 = 0 0.01;
p2 = 1 0;
p3 = 1 0.01;
p4 = 1.0045 0;
p5 = 1.0045 0.01;
p6 = 2.0045 0;
p7 = 2.0045 0.01;
n1 = 1;
n2 = 100;
tol = 1.E-5;
c1 = p0 d n1 p1;
bar1 = c1 tran n2 p2;
bar2 = bar1 plus (1.0045 0);
bars = bar1 et bar2;
elim tol (bars et p0 et p1 et p2 et p3 et p4 et p5 et p6 et p7);
con1 = bar1 elem cont p2;
con2 = bar2 elem cont p4;
mesh = bars;
tass mesh;
sauv form mesh;
trac qual mesh;
trac qual (con1 et con2);
fin;
```

bar202.epx

```
BAR202
ECHO
!CONV win
CAST mesh
CPLA LAGR
GEOM CAR1 mesh TERM
COMP COUL VERT LECT tous TERM
MATE LINE RO 8000.0 YOUN 1.E12 NU 0.0
          LECT tous TERM
OPTI PINS STAT
         !ASN
LINK COUP PINB BODY MLEV 2 LECT con1 TERM
               BODY MLEV 2 LECT con2 TERM
INIT VITE 1  50 LECT bar1 TERM
     VITE 1 -50 LECT bar2 TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA CONT ECRO TFRE 0.4E-3
     FICH ALIC FREQ 1
OPTI NOTE
 CSTA 0.5E0
 LOG 1
CALC TINI 0. TEND 0.4E-3
*=================================================================
!PLAY
!CAME   1 EYE   5.00000E-01  2.21000E+00  1.21286E+01
!!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
!         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
!         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
!         UP    0.00000E+00  1.00000E+00  0.00000E+00
!         FOV   2.48819E+01
!FREQ 2
!GO
!SCEN GEOM NAVI FREE
!          FACE HFRO
!          PINB PARE
!     COLO PAPE
!SLER CAM1 1 NFRA 1
!TRAC OFFS FICH BMP REND
!SCEN GEOM NAVI FREE
!          FACE HFRO
!          PINB NASN PASN
!     COLO PAPE
!SLER CAM1 1 NFRA 1
!TRAC OFFS FICH BMP REND
!SCEN GEOM NAVI FREE
!          FACE HFRO
!          PINB NORM
!     COLO PAPE
!SLER CAM1 1 NFRA 1
!TRAC OFFS FICH BMP REND
!FREQ 30
!GO
!ENDPLAY
*=================================================================
SUIT
Post-treatment (time curves from alice file)
ECHO
RESU ALIC GARD PSCR
SORT GRAP
AXTE 1.0 'Time [s]'
COUR  1 'dx_p0' DEPL COMP 1 NOEU LECT p0 TERM
COUR  2 'dx_p2' DEPL COMP 1 NOEU LECT p2 TERM
COUR  3 'dx_p4' DEPL COMP 1 NOEU LECT p4 TERM
COUR  4 'dx_p6' DEPL COMP 1 NOEU LECT p6 TERM
COUR  5 'vx_p0' VITE COMP 1 NOEU LECT p0 TERM
COUR  6 'vx_p2' VITE COMP 1 NOEU LECT p2 TERM
COUR  7 'vx_p4' VITE COMP 1 NOEU LECT p4 TERM
COUR  8 'vx_p6' VITE COMP 1 NOEU LECT p6 TERM
TRAC 1 2 3 4 AXES 1.0 'DISPL. [M]' YZER
TRAC 5 6 7 8 AXES 1.0 'VELOC. [M/S]' YZER
*=================================================================
FIN
```

bar202a.epx

```
BAR202 Post-treatment (animation from alice file)
ECHO
RESU ALIC 'bar202.ali' GARD PSCR
SORT VISU NSTO 1
*=================================================================
PLAY
CAME   1 EYE   3.00000E+00  3.20711E+00  2.90257E+01
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
SCEN GEOM NAVI FREE
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH AVI NOCL NFTO 1001 FPS 25 KFRE 10 COMP -1 REND
FREQ 1
GOTR LOOP 999 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDPLAY
*=================================================================
FIN
```

bar203.dgibi

```
opti echo 1;
opti dime 2 elem qua4;
opti sauv form 'bar203.msh';
opti trac psc ftra 'bar203_mesh.ps';
p0 = 0 0;
p1 = 0 0.01;
p2 = 1 0;
p3 = 1 0.01;
p4 = 1.0045 0;
p5 = 1.0045 0.01;
p6 = 2.0045 0;
p7 = 2.0045 0.01;
n1 = 1;
n2 = 100;
tol = 1.E-5;
c1 = p0 d n1 p1;
bar1 = c1 tran n2 p2;
bar2 = bar1 plus (1.0045 0);
bars = bar1 et bar2;
elim tol (bars et p0 et p1 et p2 et p3 et p4 et p5 et p6 et p7);
con1 = bar1 elem cont p2;
con2 = bar2 elem cont p4;
mesh = bars;
tass mesh;
sauv form mesh;
trac qual mesh;
trac qual (con1 et con2);
fin;
```

bar203.epx

```
BAR203
ECHO
!CONV win
CAST mesh
CPLA LAGR
GEOM CAR1 mesh TERM
COMP COUL VERT LECT tous TERM
MATE LINE RO 8000.0 YOUN 1.E12 NU 0.0
          LECT tous TERM
OPTI PINS STAT
         !ASN
LINK COUP PINB BODY MLEV 4 LECT con1 TERM
               BODY MLEV 4 LECT con2 TERM
INIT VITE 1  50 LECT bar1 TERM
     VITE 1 -50 LECT bar2 TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA CONT ECRO TFRE 0.4E-3
     FICH ALIC FREQ 1
OPTI NOTE
 CSTA 0.5E0
 LOG 1
CALC TINI 0. TEND 0.4E-3
*=================================================================
!PLAY
!CAME   1 EYE   5.00000E-01  2.21000E+00  1.21286E+01
!!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
!         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
!         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
!         UP    0.00000E+00  1.00000E+00  0.00000E+00
!         FOV   2.48819E+01
!FREQ 2
!GO
!SCEN GEOM NAVI FREE
!          FACE HFRO
```

```
!          PINB PARE
!     COLO PAPE
!SLER CAM1 1 NFRA 1
!TRAC OFFS FICH BMP REND
!SCEN GEOM NAVI FREE
!          FACE HFRO
!          PINB NASN PASN
!     COLO PAPE
!SLER CAM1 1 NFRA 1
!TRAC OFFS FICH BMP REND
!SCEN GEOM NAVI FREE
!          FACE HFRO
!          PINB NORM
!     COLO PAPE
!SLER CAM1 1 NFRA 1
!TRAC OFFS FICH BMP REND
!FREQ 30
!GO
!ENDPLAY
*=================================================================
SUIT
Post-treatment (time curves from alice file)
ECHO
RESU ALIC GARD PSCR
SORT GRAP
AXTE 1.0 'Time [s]'
COUR  1 'dx_p0' DEPL COMP 1 NOEU LECT p0 TERM
COUR  2 'dx_p2' DEPL COMP 1 NOEU LECT p2 TERM
COUR  3 'dx_p4' DEPL COMP 1 NOEU LECT p4 TERM
COUR  4 'dx_p6' DEPL COMP 1 NOEU LECT p6 TERM
COUR  5 'vx_p0' VITE COMP 1 NOEU LECT p0 TERM
COUR  6 'vx_p2' VITE COMP 1 NOEU LECT p2 TERM
COUR  7 'vx_p4' VITE COMP 1 NOEU LECT p4 TERM
COUR  8 'vx_p6' VITE COMP 1 NOEU LECT p6 TERM
TRAC 1 2 3 4 AXES 1.0 'DISPL. [M]' YZER
TRAC 5 6 7 8 AXES 1.0 'VELOC. [M/S]' YZER
*=================================================================
FIN
```

bar203a.epx

```
BAR203 Post-treatment (animation from alice file)
ECHO
RESU ALIC 'bar203.ali' GARD PSCR
SORT VISU NSTO 1
*=================================================================
PLAY
CAME   1 EYE   3.00000E+00  3.20711E+00  2.90257E+01
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
SCEN GEOM NAVI FREE
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH AVI NOCL NFTO 1001 FPS 25 KFRE 10 COMP -1 REND
FREQ 1
GOTR LOOP 999 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDPLAY
*=================================================================
FIN
```

bar204.dgibi

```
opti echo 1;
opti dime 2 elem qua4;
opti sauv form 'bar204.msh';
opti trac psc ftra 'bar204_mesh.ps';
p0 = 0 0;
p1 = 0 0.01;
p2 = 1 0;
p3 = 1 0.01;
p4 = 1.0045 0;
p5 = 1.0045 0.01;
p6 = 2.0045 0;
p7 = 2.0045 0.01;
n1 = 1;
n2 = 100;
tol = 1.E-5;
c1 = p0 d n1 p1;
bar1 = c1 tran n2 p2;
bar2 = bar1 plus (1.0045 0);
bars = bar1 et bar2;
elim tol (bars et p0 et p1 et p2 et p3 et p4 et p5 et p6 et p7);
con1 = bar1 elem cont p2;
con2 = bar2 elem cont p4;
mesh = bars;
tass mesh;
sauv form mesh;
trac qual mesh;
trac qual (con1 et con2);
fin;
```

bar204.epx

```
BAR204
ECHO
!CONV win
CAST mesh
CPLA LAGR
GEOM CAR1 mesh TERM
COMP COUL VERT LECT tous TERM
MATE LINE RO 8000.0 YOUN 1.E12 NU 0.0
          LECT tous TERM
OPTI PINS STAT
          ASN
          NORB
LINK DECO PINB PENA SFAC 1.0
               BODY LECT con1 TERM
               BODY LECT con2 TERM
INIT VITE 1  50 LECT bar1 TERM
     VITE 1 -50 LECT bar2 TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA CONT ECRO TFRE 0.4E-3
     FICH ALIC FREQ 1
OPTI NOTE
 CSTA 0.5E0
 LOG 1
CALC TINI 0. TEND 0.4E-3
*=================================================================
!PLAY
!CAME   1 EYE   5.00000E-01  2.21000E+00  1.21286E+01
!!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
!         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
!         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
!         UP    0.00000E+00  1.00000E+00  0.00000E+00
!         FOV   2.48819E+01
!FREQ 2
!GO
!SCEN GEOM NAVI FREE
!          FACE HFRO
!          PINB PARE
!     COLO PAPE
!SLER CAM1 1 NFRA 1
!TRAC OFFS FICH BMP REND
!SCEN GEOM NAVI FREE
!          FACE HFRO
!          PINB NASN PASN
!     COLO PAPE
!SLER CAM1 1 NFRA 1
!TRAC OFFS FICH BMP REND
!SCEN GEOM NAVI FREE
!          FACE HFRO
!          PINB NORM
!     COLO PAPE
!SLER CAM1 1 NFRA 1
!TRAC OFFS FICH BMP REND
!FREQ 30
!GO
!ENDPLAY
*=================================================================
SUIT
Post-treatment (time curves from alice file)
ECHO
RESU ALIC GARD PSCR
SORT GRAP
AXTE 1.0 'Time [s]'
COUR  1 'dx_p0' DEPL COMP 1 NOEU LECT p0 TERM
COUR  2 'dx_p2' DEPL COMP 1 NOEU LECT p2 TERM
COUR  3 'dx_p4' DEPL COMP 1 NOEU LECT p4 TERM
COUR  4 'dx_p6' DEPL COMP 1 NOEU LECT p6 TERM
COUR  5 'vx_p0' VITE COMP 1 NOEU LECT p0 TERM
COUR  6 'vx_p2' VITE COMP 1 NOEU LECT p2 TERM
COUR  7 'vx_p4' VITE COMP 1 NOEU LECT p4 TERM
COUR  8 'vx_p6' VITE COMP 1 NOEU LECT p6 TERM
TRAC 1 2 3 4 AXES 1.0 'DISPL. [M]' YZER
TRAC 5 6 7 8 AXES 1.0 'VELOC. [M/S]' YZER
*=================================================================
FIN
```

bar204a.epx

```
BAR204 Post-treatment (animation from alice file)
ECHO
RESU ALIC 'bar204.ali' GARD PSCR
SORT VISU NSTO 1
*=================================================================
PLAY
CAME   1 EYE   3.00000E+00  3.20711E+00  2.90257E+01
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
SCEN GEOM NAVI FREE
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH AVI NOCL NFTO 1001 FPS 25 KFRE 10 COMP -1 REND
FREQ 1
GOTR LOOP 999 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDPLAY
*=================================================================
FIN
```

bar205.dgibi

```
opti echo 1;
opti dime 2 elem qua4;
opti sauv form 'bar205.msh';
opti trac psc ftra 'bar205_mesh.ps';
p0 = 0 0;
p1 = 0 0.01;
p2 = 1 0;
p3 = 1 0.01;
p4 = 1.0045 0;
p5 = 1.0045 0.01;
p6 = 2.0045 0;
p7 = 2.0045 0.01;
n1 = 1;
n2 = 100;
tol = 1.E-5;
c1 = p0 d n1 p1;
bar1 = c1 tran n2 p2;
bar2 = bar1 plus (1.0045 0);
bars = bar1 et bar2;
elim tol (bars et p0 et p1 et p2 et p3 et p4 et p5 et p6 et p7);
con1 = bar1 elem cont p2;
con2 = bar2 elem cont p4;
mesh = bars;
tass mesh;
```

```
sauv form mesh;
trac qual mesh;
trac qual (con1 et con2);
fin;
```

bar205.epx

```
BAR205
ECHO
!CONV win
CAST mesh
CPLA LAGR
GEOM CAR1 mesh TERM
COMP COUL VERT LECT tous TERM
MATE LINE RO 8000.0 YOUN 1.E12 NU 0.0
          LECT tous TERM
OPTI PINS STAT
          ASN
          NORB
LINK DECO PINB PENA SFAC 1.0
               BODY MLEV 2 LECT con1 TERM
               BODY MLEV 2 LECT con2 TERM
INIT VITE 1  50 LECT bar1 TERM
     VITE 1 -50 LECT bar2 TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA CONT ECRO TFRE 0.4E-3
     FICH ALIC FREQ 1
OPTI NOTE
 CSTA 0.5E0
 LOG 1
CALC TINI 0. TEND 0.4E-3
*=================================================================
!PLAY
!CAME   1 EYE   5.00000E-01  2.21000E+00  1.21286E+01
!!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
!         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
!         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
!         UP    0.00000E+00  1.00000E+00  0.00000E+00
!         FOV   2.48819E+01
!FREQ 2
!GO
!SCEN GEOM NAVI FREE
!          FACE HFRO
!          PINB PARE
!     COLO PAPE
!SLER CAM1 1 NFRA 1
!TRAC OFFS FICH BMP REND
!SCEN GEOM NAVI FREE
!          FACE HFRO
!          PINB NASN PASN
!     COLO PAPE
!SLER CAM1 1 NFRA 1
!TRAC OFFS FICH BMP REND
!SCEN GEOM NAVI FREE
!          FACE HFRO
!          PINB NORM
!     COLO PAPE
!SLER CAM1 1 NFRA 1
!TRAC OFFS FICH BMP REND
!FREQ 30
!GO
!ENDPLAY
*=================================================================
SUIT
Post-treatment (time curves from alice file)
ECHO
RESU ALIC GARD PSCR
SORT GRAP
AXTE 1.0 'Time [s]'
COUR  1 'dx_p0' DEPL COMP 1 NOEU LECT p0 TERM
COUR  2 'dx_p2' DEPL COMP 1 NOEU LECT p2 TERM
COUR  3 'dx_p4' DEPL COMP 1 NOEU LECT p4 TERM
COUR  4 'dx_p6' DEPL COMP 1 NOEU LECT p6 TERM
COUR  5 'vx_p0' VITE COMP 1 NOEU LECT p0 TERM
COUR  6 'vx_p2' VITE COMP 1 NOEU LECT p2 TERM
COUR  7 'vx_p4' VITE COMP 1 NOEU LECT p4 TERM
COUR  8 'vx_p6' VITE COMP 1 NOEU LECT p6 TERM
TRAC 1 2 3 4 AXES 1.0 'DISPL. [M]' YZER
TRAC 5 6 7 8 AXES 1.0 'VELOC. [M/S]' YZER
*=================================================================
FIN
```

bar205a.epx

```
BAR205 Post-treatment (animation from alice file)
ECHO
RESU ALIC 'bar205.ali' GARD PSCR
SORT VISU NSTO 1
*=================================================================
PLAY
CAME   1 EYE   3.00000E+00  3.20711E+00  2.90257E+01
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
SCEN GEOM NAVI FREE
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH AVI NOCL NFTO 1001 FPS 25 KFRE 10 COMP -1 REND
FREQ 1
GOTR LOOP 999 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDPLAY
*=================================================================
FIN
```

bar206.dgibi

```
opti echo 1;
opti dime 2 elem qua4;
opti sauv form 'bar206.msh';
opti trac psc ftra 'bar206_mesh.ps';
p0 = 0 0;
p1 = 0 0.01;
p2 = 1 0;
p3 = 1 0.01;
p4 = 1.0045 0;
p5 = 1.0045 0.01;
p6 = 2.0045 0;
p7 = 2.0045 0.01;
n1 = 1;
n2 = 100;
tol = 1.E-5;
c1 = p0 d n1 p1;
bar1 = c1 tran n2 p2;
bar2 = bar1 plus (1.0045 0);
bars = bar1 et bar2;
elim tol (bars et p0 et p1 et p2 et p3 et p4 et p5 et p6 et p7);
con1 = bar1 elem cont p2;
con2 = bar2 elem cont p4;
mesh = bars;
tass mesh;
sauv form mesh;
trac qual mesh;
trac qual (con1 et con2);
fin;
```

bar206.epx

```
BAR206
ECHO
!CONV win
CAST mesh
CPLA LAGR
GEOM CAR1 mesh TERM
COMP COUL VERT LECT tous TERM
MATE LINE RO 8000.0 YOUN 1.E12 NU 0.0
          LECT tous TERM
OPTI PINS STAT
          ASN
LINK DECO PINB PENA SFAC 1.0
               BODY MLEV 4 LECT con1 TERM
               BODY MLEV 4 LECT con2 TERM
INIT VITE 1  50 LECT bar1 TERM
     VITE 1 -50 LECT bar2 TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA CONT ECRO TFRE 0.4E-3
     FICH ALIC FREQ 1
OPTI NOTE
 CSTA 0.5E0
 LOG 1
CALC TINI 0. TEND 0.4E-3
*=================================================================
!PLAY
!CAME   1 EYE   5.00000E-01  2.21000E+00  1.21286E+01
!!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
!         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
!         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
!         UP    0.00000E+00  1.00000E+00  0.00000E+00
!         FOV   2.48819E+01
!FREQ 2
!GO
!SCEN GEOM NAVI FREE
!          FACE HFRO
!          PINB PARE
!     COLO PAPE
!SLER CAM1 1 NFRA 1
!TRAC OFFS FICH BMP REND
!SCEN GEOM NAVI FREE
!          FACE HFRO
!          PINB NASN PASN
!     COLO PAPE
!SLER CAM1 1 NFRA 1
!TRAC OFFS FICH BMP REND
!SCEN GEOM NAVI FREE
!          FACE HFRO
!          PINB NORM
!     COLO PAPE
!SLER CAM1 1 NFRA 1
!TRAC OFFS FICH BMP REND
!FREQ 30
!GO
!ENDPLAY
*=================================================================
SUIT
Post-treatment (time curves from alice file)
ECHO
RESU ALIC GARD PSCR
SORT GRAP
AXTE 1.0 'Time [s]'
COUR  1 'dx_p0' DEPL COMP 1 NOEU LECT p0 TERM
COUR  2 'dx_p2' DEPL COMP 1 NOEU LECT p2 TERM
COUR  3 'dx_p4' DEPL COMP 1 NOEU LECT p4 TERM
COUR  4 'dx_p6' DEPL COMP 1 NOEU LECT p6 TERM
COUR  5 'vx_p0' VITE COMP 1 NOEU LECT p0 TERM
COUR  6 'vx_p2' VITE COMP 1 NOEU LECT p2 TERM
COUR  7 'vx_p4' VITE COMP 1 NOEU LECT p4 TERM
COUR  8 'vx_p6' VITE COMP 1 NOEU LECT p6 TERM
TRAC 1 2 3 4 AXES 1.0 'DISPL. [M]' YZER
TRAC 5 6 7 8 AXES 1.0 'VELOC. [M/S]' YZER
*=================================================================
FIN
```

bar206a.epx

```
BAR206 Post-treatment (animation from alice file)
ECHO
RESU ALIC 'bar206.ali' GARD PSCR
SORT VISU NSTO 1
*=================================================================
PLAY
CAME   1 EYE   3.00000E+00  3.20711E+00  2.90257E+01
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
```

```
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
SCEN GEOM NAVI FREE
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH AVI NOCL NFTO 1001 FPS 25 KFRE 10 COMP -1 REND
FREQ 1
GOTR LOOP 999 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDPLAY
*==================================================================
FIN
```

carn07.epx

```
CARN07
ECHO
!CONV WIN
CAST mesh
ALE DPLA
DIME NALE 1 NBLE 3180 TERM
GEOM Q42 sur4 TERM
COMP EPAI 1. LECT sur4 TERM
     NGRO 3
       'noeuds_sym'  LECT nodlag TERM COND X GT 14.95
       'noeuds_haut' LECT nodlag TERM COND Y GT 16.95
       'noeuds_bas'  LECT nodlag TERM COND Y LT  0.05
     COUL VERT LECT sur4 TERM
GRIL LAGR LECT nodlag TERM
     AUTO AUTR
MATE VM23 RO 0.00825 YOUN 197600.0 NU 0.29 ELAS 222.35
          TRAC 39
            222.35           0.00112525303643725
            228.192645571222 0.00117982108082602
            230.758344471793 0.001217805387003
            234.450726613396 0.00128649153144431
            239.764555869277 0.00141338337990525
            243.897432663364 0.00153429874829638
            247.411863292444 0.00165208432840306
            250.528021456793 0.00176785435959915
            253.359622970924 0.00188218432677593
            258.417356319849 0.00210778014331907
            262.901922592959 0.00233047531676599
            272.525831528572 0.00287917930935512
            280.709612952488 0.00342059520724944
            306.337249091751 0.00555028972212425
            326.269365384357 0.00765116075599371
            358.247494301277 0.0118129933922129
            390.499112131117 0.0169762100816352
            417.924578503771 0.0221150029276507
            464.338948360747 0.0323498934633641
            503.807844043327 0.0425496348382759
            538.803931461142 0.0527267405438317
            570.604298738599 0.0628876735766123
            599.971697212392 0.0730362940142328
            627.405291846074 0.0831751279951724
            653.253096704044 0.0933059367242108
            677.76931865327  0.103430006673347
            734.395540393597 0.128716576621425
            785.850853879881 0.153976978005465
            833.369446403083 0.179217456712566
            877.759003278812 0.204442100219022
            959.251671749752 0.254854512508855
            1033.30271513801 0.305229264752723
            1165.57077256009 0.405898637512956
            1282.84956691385 0.506492153678714
            1389.41887249881 0.607031472026816
            1579.77020835558 0.807994788503824
            1748.55          1.00884893724696
            2418.75489508018 2.0122406624245
            5336.94734929116 10.0270088428608
          LECT sur4 TERM
OPTI PINS GRID DGRI EQVF ASN NORB
LINK DECO
     BLOQ 12 LECT noeuds_bas TERM
     BLOQ 1  LECT noeuds_sym TERM
     PINB PENA SFAC 1.0
          SELF DMIN 0.1 LECT elepin TERM
CHAR 1 FACT 2
     DEPL 2 -10 LECT noeuds_haut TERM
     TABL 3 0 0 1 1 2 2
INIT VITE 2 -10 LECT noeuds_haut TERM
ECRI DEPL TFRE 0.01 POIN LECT 4460 TERM
     FICH SPLI ALIC TFRE 0.005
REGI 'HAUT' TOUT POIN LECT noeuds_haut TERM
     'BAS'  TOUT POIN LECT noeuds_bas TERM
OPTI PAS AUTO
     LOG 1
CALC TINI 0.0 TFIN 1.0
FIN
```

carn07a.epx

```
Post treatment (visualization from alice file)
ECHO
RESU SPLI ALIC 'carn07.ali' GARD PSCR
OPTI PRIN
SORT VISU NSTO 1
*====================================================================
PLAY
CAME   1 EYE   5.56399E+00  8.50620E+00  4.84072E+01
!        Q      1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  5.56399E+00  8.50620E+00  0.00000E+00
!RSPHERE:  1.27387E+01
!RADIUS :  4.84072E+01
!ASPECT :  1.00000E+00
!NEAR   :  3.56685E+01
!FAR    :  7.38847E+01
SCEN GEOM NAVI FREE
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH AVI NOCL NFTO 201 FPS 25 KFRE 10 COMP -1 REND
FREQ 1
GOTR LOOP 199 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDPLAY
*====================================================================
FIN
```

carn07b.epx

```
Post treatment (visualization from alice file)
ECHO
RESU SPLI ALIC 'carn07.ali' GARD PSCR
OPTI PRIN
SORT VISU NSTO 1
*====================================================================
PLAY
CAME   1 EYE   5.56399E+00  8.50620E+00  4.84072E+01
!        Q      1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  5.56399E+00  8.50620E+00  0.00000E+00
!RSPHERE:  1.27387E+01
!RADIUS :  4.84072E+01
!ASPECT :  1.00000E+00
!NEAR   :  3.56685E+01
!FAR    :  7.38847E+01
SCEN GEOM NAVI FREE
          LINE HEOU SFRE
     ISO FILL FIEL ECRO 3 SCAL USER PROG 0.05 PAS 0.05 0.7 TERM
     TEXT ISCA
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH AVI NOCL NFTO 201 FPS 25 KFRE 10 COMP -1 REND
FREQ 1
GOTR LOOP 199 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDPLAY
*====================================================================
FIN
```

carn07c.epx

```
Post treatment (visualization from alice file)
ECHO
RESU SPLI ALIC 'carn07.ali' GARD PSCR
OPTI PRIN
SORT VISU NSTO 1
*====================================================================
PLAY
CAME   1 EYE   5.56399E+00  8.50620E+00  4.84072E+01
!        Q      1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  5.56399E+00  8.50620E+00  0.00000E+00
!RSPHERE:  1.27387E+01
!RADIUS :  4.84072E+01
!ASPECT :  1.00000E+00
!NEAR   :  3.56685E+01
!FAR    :  7.38847E+01
SCEN GEOM NAVI FREE
         !LINE HEOU SFRE
          FACE HFRO
          PINB CDES
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH AVI NOCL NFTO 201 FPS 25 KFRE 10 COMP -1 REND
FREQ 1
GOTR LOOP 199 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDPLAY
*====================================================================
FIN
```

carn07d.epx

```
Post treatment (visualization from alice file)
ECHO
RESU SPLI ALIC 'carn07.ali' GARD PSCR
OPTI PRIN
SORT GRAP
AXTE 1.0 'Time [s]'
COUR 1 'dy_350' DEPL COMP 2 NOEU LECT 350 TERM
COUR 2 'dy' MULC 1 -1.0
COUR 3 'fyh' FEXT COMP 2 ZONE LECT noeuds_haut TERM
COUR 4 'fyb' FDEC COMP 2 ZONE LECT noeuds_bas TERM
COUR 5 'fyhm1' MULC 3 -1.0
TRAC 2 AXES 1.0 'DISPL. [MM]' YZER XGRD YGRD
TRAC 4 5 AXES 1.0 'FORCE [N]' YZER XGRD YGRD
COLO NOIR ROUG
LIST 3 4 AXES 1.0 'FORCE [N]'
TRAC 4 AXES 1.0 'FORCE [N]' XAXE 2 1.0 'DISPL. [MM]' YZER XGRD YGRD
!RCOU 44 'fyb' FICH 'car401d.pun' RENA 'fyb_401'
!RCOU 54 'fyb' FICH 'car102d.pun' RENA 'fyb_102'
!TRAC 4 44 54 AXES 1.0 'FORCE [N]' YZER XGRD YGRD
!COLO NOIR ROUG VERT
!TRAC 4 54 AXES 1.0 'FORCE [N]' YZER XGRD YGRD
```

```
!COLO NOIR VERT
RCOU 44 'fyb' FICH 'cara07d.pun' RENA 'fyb_A07'
TRAC 4 44 AXES 1.0 'FORCE [N]' YZER XGRD YGRD
COLO NOIR VERT
FIN
```

carn07e.epx

```
Post treatment (visualization from alice file)
ECHO
RESU SPLI ALIC 'carn07.ali' GARD PSCR
OPTI PRIN
SORT GRAP
AXTE 1.0 'Time [s]'
COUR 1 'dyh' DEPL COMP 2 NOEU LECT 350 TERM
COUR 2 'dy' MULC 1 -1.0
COUR 3 'fyh' FEXT COMP 2 ZONE LECT noeuds_haut TERM
COUR 4 'fy' MULC 3 -1.0
COUR 5 'fyb' FDEC COMP 2 ZONE LECT noeuds_bas TERM
TRAC 4 AXES 1.0 'FORCE [N]' XAXE 2 1.0 'DISPL. [MM]' YZER XGRD YGRD
LIST 4 AXES 1.0 'FORCE [N]' XAXE 2 1.0 'DISPL. [MM]' YZER XGRD YGRD
!RCOU 24 'fy' FICH 'cara02e.pun' RENA 'fy_02'
!RCOU 44 'fy' FICH 'cara04e.pun' RENA 'fy_04'
!RCOU 54 'fy' FICH 'cara05e.pun' RENA 'fy_05'
!RCOU 64 'fy' FICH 'cara06e.pun' RENA 'fy_06'
!RCOU 74 'fy' FICH 'cara07e.pun' RENA 'fy_07'
!TRAC 24 44 54 64 74 AXES 1.0 'FORCE [N]' YZER XGRD YGRD
!COLO NOIR ROUG VERT TURQ ROSE
!TRAC 2 AXES 1.0 'DISPL. [MM]' YZER XGRD YGRD
!TRAC 4 5 AXES 1.0 'FORCE [N]' YZER XGRD YGRD
!COLO NOIR ROUG
FIN
```

caro07.epx

```
CARO07
ECHO
!CONV WIN
CAST mesh
ALE DPLA
DIME NALE 1 NBLE 3180 TERM
GEOM Q42 sur4 TERM
COMP EPAI 1. LECT sur4 TERM
     NGRO 3
       'noeuds_sym'  LECT nodlag TERM COND X GT 14.95
       'noeuds_haut' LECT nodlag TERM COND Y GT 16.95
       'noeuds_bas'  LECT nodlag TERM COND Y LT  0.05
     COUL VERT LECT sur4 TERM
GRIL LAGR LECT nodlag TERM
     AUTO AUTR
MATE VM23 RO 0.00825 YOUN 197600.0 NU 0.29 ELAS 222.35
        TRAC 39
          222.35           0.00112525303643725
          228.192645571222 0.00117982108082602
          230.758344471793 0.001217805387003
          234.450726613396 0.00128649153144431
          239.764555869277 0.00141338337990525
          243.897432663364 0.00153429874829638
          247.411863292444 0.00165208432840306
          250.528021456793 0.00176785435959915
          253.359622970924 0.00188218432677593
          258.417356319849 0.00210778014331907
          262.901922592959 0.00233047531676599
          272.525831528572 0.00287917930935512
          280.709612952488 0.00342059520724944
          306.337249091751 0.00555028972212425
          326.269365384357 0.00765116075599371
          358.247494301277 0.0118129933922129
          390.499112131117 0.0169762100816352
          417.924578503771 0.0221150029276507
          464.338948360747 0.0323498934633641
          503.807844043327 0.0425496348382759
          538.803931461142 0.0527267405438317
          570.604298738599 0.0628876735766123
          599.971697212392 0.0730362940142328
          627.405291846074 0.0831751279951724
          653.253096704044 0.0933059367242108
          677.76931865327  0.103430006673347
          734.395540393597 0.128716576621425
          785.850853879881 0.153976978005465
          833.369446403083 0.179217456712566
          877.759003278812 0.204442100219022
          959.251671749752 0.254854512508855
          1033.30271513801 0.305229264752723
          1165.57077256009 0.405898637512956
          1282.84956691385 0.506492153678714
          1389.41887249881 0.607031472026816
          1579.77020835558 0.807994788503824
          1748.55          1.00884893724696
          2418.75489508018 2.0122406624245
          5336.94734929116 10.0270088428608
        LECT sur4 TERM
OPTI PINS GRID DGRI EQVF ASN NORB
LINK COUP SPLT NONE
     BLOQ 12 LECT noeuds_bas TERM
     BLOQ 1  LECT noeuds_sym TERM
     VITE 2 -10 FONC 1 LECT noeuds_haut TERM
     PINB SELF DMIN 0.1 LECT elepin TERM
FONC 1 TABLE 2 0 1 100 1
INIT VITE 2 -10 LECT noeuds_haut TERM
ECRI DEPL TFRE 0.01 POIN LECT 4460 TERM
     FICH SPLI ALIC TFRE 0.005
REGI 'HAUT' TOUT POIN LECT noeuds_haut TERM
     'BAS'  TOUT POIN LECT noeuds_bas TERM
OPTI PAS AUTO
     LOG 1
CALC TINI 0.0 TFIN 1.0
FIN
```

caro07a.epx

```
Post treatment (visualization from alice file)
ECHO
RESU SPLI ALIC 'caro07.ali' GARD PSCR
OPTI PRIN
SORT VISU NSTO 1
*=====================================================================
PLAY
CAME   1 EYE   5.56399E+00  8.50620E+00  4.84072E+01
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  5.56399E+00  8.50620E+00  0.00000E+00
!RSPHERE:  1.27387E+01
!RADIUS :  4.84072E+01
!ASPECT :  1.00000E+00
!NEAR   :  3.56685E+01
!FAR    :  7.38847E+01
SCEN GEOM NAVI FREE
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH AVI NOCL NFTO 201 FPS 25 KFRE 10 COMP -1 REND
FREQ 1
GOTR LOOP 199 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDPLAY
*=====================================================================
FIN
```

caro07b.epx

```
Post treatment (visualization from alice file)
ECHO
RESU SPLI ALIC 'caro07.ali' GARD PSCR
OPTI PRIN
SORT VISU NSTO 1
*=====================================================================
PLAY
CAME   1 EYE   5.56399E+00  8.50620E+00  4.84072E+01
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  5.56399E+00  8.50620E+00  0.00000E+00
!RSPHERE:  1.27387E+01
!RADIUS :  4.84072E+01
!ASPECT :  1.00000E+00
!NEAR   :  3.56685E+01
!FAR    :  7.38847E+01
SCEN GEOM NAVI FREE
          LINE HEOU SFRE
     ISO FILL FIEL ECRO 3 SCAL USER PROG 0.05 PAS 0.05 0.7 TERM
     TEXT ISCA
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH AVI NOCL NFTO 201 FPS 25 KFRE 10 COMP -1 REND
FREQ 1
GOTR LOOP 199 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDPLAY
*=====================================================================
FIN
```

caro07c.epx

```
Post treatment (visualization from alice file)
ECHO
RESU SPLI ALIC 'caro07.ali' GARD PSCR
OPTI PRIN
SORT VISU NSTO 1
*=====================================================================
PLAY
CAME   1 EYE   5.56399E+00  8.50620E+00  4.84072E+01
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  5.56399E+00  8.50620E+00  0.00000E+00
!RSPHERE:  1.27387E+01
!RADIUS :  4.84072E+01
!ASPECT :  1.00000E+00
!NEAR   :  3.56685E+01
!FAR    :  7.38847E+01
SCEN GEOM NAVI FREE
         !LINE HEOU SFRE
          FACE HFRO
          PINB CDES
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH AVI NOCL NFTO 201 FPS 25 KFRE 10 COMP -1 REND
FREQ 1
GOTR LOOP 199 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDPLAY
*=====================================================================
FIN
```

caro07d.epx

```
Post treatment (visualization from alice file)
ECHO
RESU SPLI ALIC 'caro07.ali' GARD PSCR
OPTI PRIN
SORT GRAP
AXTE 1.0 'Time [ms]'
COUR 1 'dy_350' DEPL COMP 2 NOEU LECT 350 TERM
COUR 2 'dy' MULC 1 -1.0
COUR 3 'fyh' FLIA COMP 2 ZONE LECT noeuds_haut TERM
COUR 4 'fyb' FLIA COMP 2 ZONE LECT noeuds_bas TERM
TRAC 2 AXES 1.0 'DISPL. [MM]' YZER XGRD YGRD
TRAC 3 4 AXES 1.0 'FORCE [N]'
LIST 3 4 AXES 1.0 'FORCE [N]'
TRAC 4 AXES 1.0 'FORCE [N]' XAXE 2 1.0 'DISPL. [MM]' YZER XGRD YGRD
!RCOU 44 'fyb' FICH 'car401d.pun' RENA 'fyb_401'
!RCOU 54 'fyb' FICH 'car102d.pun' RENA 'fyb_102'
!TRAC 4 44 54 AXES 1.0 'FORCE [N]' YZER XGRD YGRD
!COLO NOIR ROUG VERT
!TRAC 4 54 AXES 1.0 'FORCE [N]' YZER XGRD YGRD
!COLO NOIR VERT
RCOU 44 'fyb' FICH 'cara07d.pun' RENA 'fyb_A07'
RCOU 54 'fyb' FICH 'carn07d.pun' RENA 'fyb_N07'
TRAC 4 44 54 AXES 1.0 'FORCE [N]' YZER XGRD YGRD
COLO NOIR VERT ROUG
FIN
```

caro07e.epx

```
Post treatment (visualization from alice file)
ECHO
RESU SPLI ALIC 'caro07.ali' GARD PSCR
OPTI PRIN
SORT GRAP
AXTE 1.0 'Time [s]'
COUR 1 'dyh' DEPL COMP 2 NOEU LECT 350 TERM
COUR 2 'dy' MULC 1 -1.0
COUR 3 'fyh' FLIA COMP 2 ZONE LECT noeuds_haut TERM
COUR 4 'fy' MULC 3 -1.0
COUR 5 'fyb' FLIA COMP 2 ZONE LECT noeuds_bas TERM
TRAC 4 AXES 1.0 'FORCE [N]' XAXE 2 1.0 'DISPL. [MM]' YZER XGRD YGRD
LIST 4 AXES 1.0 'FORCE [N]' XAXE 2 1.0 'DISPL. [MM]' YZER XGRD YGRD
RCOU 24 'fy' FICH 'cara02e.pun' RENA 'fy_02'
RCOU 44 'fy' FICH 'cara04e.pun' RENA 'fy_04'
RCOU 54 'fy' FICH 'cara05e.pun' RENA 'fy_05'
RCOU 64 'fy' FICH 'cara06e.pun' RENA 'fy_06'
RCOU 74 'fy' FICH 'caro07e.pun' RENA 'fy_07'
TRAC 24 44 54 64 74 AXES 1.0 'FORCE [N]' YZER XGRD YGRD
COLO NOIR ROUG VERT TURQ ROSE
TRAC 2 AXES 1.0 'DISPL. [MM]' YZER XGRD YGRD
TRAC 4 5 AXES 1.0 'FORCE [N]' YZER XGRD YGRD
COLO NOIR ROUG
FIN
```

dro201.dgibi

```
opti echo 1;
opti dime 2 elem qua4;
opti sauv form 'dro201.msh';
opti trac psc ftra 'dro201_mesh.ps';
p1 = -1 0;
p2 = 0 0;
p3 = 6 0;
p4 = 7 0;
p5 = -1 -1;
p6 = 0 6;
p7 = 6 6;
p8 = 7 -1;
p9 = -1 6;
p10 = 7 6;
n1 = 2;
n2 = 12;
tol = 1.E-3;
c1 = p1 d n1 p2;
box1 = c1 tran n2 p6;
box2 = box1 plus (7 0);
c2 = p1 d n1 p5;
box3 = c2 tran (n1 + n1 + n2) (8 0);
box = box1 et box2 et box3;
elim tol (box et p1 et p2 et p3 et p4 et p5 et p6 et p7
          et p8 et p9 et p10);
c3 = (p2 plus (0 0)) d (n1 + n1) (2 0);
squa1 = c3 tran (n1 + n1) (0 2);
depl squa1 plus (1 5);
depl squa1 tour 45 (bary squa1);
c4 = (p2 plus (0 0)) d (n1 + n1 + n1 + n1) (4 0);
rect1 = c4 tran n1 (0 1);
depl rect1 plus (1.5 3);
depl rect1 tour 30 (bary rect1);
squa2 = squa1 plus (2.5 (0 - 4.6));
depl squa2 tour (0-30) (bary squa2);
mesh = box et squa1 et rect1 et squa2;
lint = p6 d n2 p2 d n2 p3 d n2 p7;
elim tol (lint et box);
lext = p9 d n2 p1 d n1 p5 d (n1 + n2 + n1) p8 d n1 p4 d n2 p10;
elim tol (lext et box);
pbox = box elem appu larg lint;
psqua1 = squa1 elem appu larg (cont squa1);
prect1 = rect1 elem appu larg (cont rect1);
psqua2 = squa2 elem appu larg (cont squa2);
tass mesh;
sauv form mesh;
trac qual mesh;
trac qual (pbox et psqua1 et prect1 et psqua2);
fin;
```

dro201.epx

```
DRO201
ECHO
!CONV win
CAST mesh
CPLA LAGR
GEOM CAR1 mesh TERM
COMP COUL VERT LECT tous TERM
MATE LINE RO 8000.0 YOUN 1.E12 NU 0.0
         LECT tous TERM
OPTI PINS STAT
         ASN
         NORB
LINK COUP BLOQ 12 LECT lext TERM
LINK DECO PINB PENA SFAC 1.0
              BODY MLEV 2 LECT pbox   TERM
              BODY MLEV 2 LECT psqua1 TERM
              BODY MLEV 2 LECT prect1 TERM
              BODY MLEV 2 LECT psqua2 TERM
CHAR CONS GRAV 0 -9.80665D0 LECT squa1 rect1 squa2 TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA CONT ECRO TFRE 1
     FICH ALIC TFRE 1.E-2
OPTI NOTE
 CSTA 0.5E0
 LOG 10000
CALCUL TINI 0. TEND 10.E0 NMAX 1000000
fin

*=================================================================
PLAY
CAME   1 EYE   5.00000E-01  2.21000E+00  1.21286E+01
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
FREQ 2
GO
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB PARE
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB NASN PASN
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB NORM
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
FREQ 30
GO
ENDPLAY
*=================================================================
SUIT
Post-treatment (time curves from alice file)
ECHO
RESU ALIC GARD PSCR
SORT GRAP
AXTE 1.0 'Time [s]'
COUR  1 'dy_4'  DEPL COMP 2 NOEU LECT  4 TERM
COUR  2 'dy_5'  DEPL COMP 2 NOEU LECT  5 TERM
COUR 11 'fy_4'  FEXT COMP 2 NOEU LECT  4 TERM
COUR 12 'fy_5'  FEXT COMP 2 NOEU LECT  5 TERM
TRAC 1 2 AXES 1.0 'DISPL. [M]' YZER
TRAC 11 12 AXES 1.0 'FEXT [N]' YZER
LIST 1 2 AXES 1.0 'DISPL. [M]' YZER
LIST 11 12 AXES 1.0 'FEXT [N]' YZER
*=================================================================
FIN
```

dro201a.epx

```
DRO201 Post-treatment (animation from alice file)
ECHO
RESU ALIC 'dro201.ali' GARD PSCR
SORT VISU NSTO 1
*=================================================================
PLAY
CAME   1 EYE   3.00000E+00  3.20711E+00  2.90257E+01
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
SCEN GEOM NAVI FREE
          PINB CDES NORM
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH AVI NOCL NFTO 1001 FPS 25 KFRE 10 COMP -1 REND
FREQ 1
GOTR LOOP 999 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDPLAY
*=================================================================
FIN
```

dro202.dgibi

```
opti echo 1;
opti dime 2 elem qua4;
opti sauv form 'dro202.msh';
opti trac psc ftra 'dro202_mesh.ps';
p1 = -1 0;
```

```
p2 = 0 0;
p3 = 6 0;
p4 = 7 0;
p5 = -1 -1;
p6 = 0 6;
p7 = 6 6;
p8 = 7 -1;
p9 = -1 6;
p10 = 7 6;
n1 = 4;
n2 = 24;
tol = 1.E-3;
c1 = p1 d n1 p2;
box1 = c1 tran n2 p6;
box2 = box1 plus (7 0);
c2 = p1 d n1 p5;
box3 = c2 tran (n1 + n1 + n2) (8 0);
box = box1 et box2 et box3;
elim tol (box et p1 et p2 et p3 et p4 et p5 et p6 et p7
          et p8 et p9 et p10);
c3 = (p2 plus (0 0)) d (n1 + n1) (2 0);
squa1 = c3 tran (n1 + n1) (0 2);
depl squa1 plus (1 5);
depl squa1 tour 45 (bary squa1);
c4 = (p2 plus (0 0)) d (n1 + n1 + n1 + n1) (4 0);
rect1 = c4 tran n1 (0 1);
depl rect1 plus (1.5 3);
depl rect1 tour 30 (bary rect1);
squa2 = squa1 plus (2.5 (0 - 4.6));
depl squa2 tour (0-30) (bary squa2);
mesh = box et squa1 et rect1 et squa2;
lint = p6 d n2 p2 d n2 p3 d n2 p7;
elim tol (lint et box);
lext = p9 d n2 p1 d n1 p5 d (n1 + n2 + n1) p8 d n1 p4 d n2 p10;
elim tol (lext et box);
pbox = box elem appu larg lint;
psqua1 = squa1 elem appu larg (cont squa1);
prect1 = rect1 elem appu larg (cont rect1);
psqua2 = squa2 elem appu larg (cont squa2);
tass mesh;
sauv form mesh;
trac qual mesh;
trac qual (pbox et psqua1 et prect1 et psqua2);
fin;
```

dro202.epx

```
DRO202
ECHO
!CONV win
CAST mesh
CPLA LAGR
GEOM CAR1 mesh TERM
COMP COUL VERT LECT tous TERM
MATE LINE RO 8000.0 YOUN 1.E12 NU 0.0
          LECT tous TERM
OPTI PINS STAT
          ASN
          NORB
LINK COUP BLOQ 12 LECT lext TERM
LINK DECO PINB PENA SFAC 1.0
               BODY MLEV 2 LECT pbox   TERM
               BODY MLEV 2 LECT psqua1 TERM
               BODY MLEV 2 LECT prect1 TERM
               BODY MLEV 2 LECT psqua2 TERM
CHAR CONS GRAV 0 -9.80665D0 LECT squa1 rect1 squa2 TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA CONT ECRO TFRE 1
     FICH ALIC TFRE 1.E-2
OPTI NOTE
 CSTA 0.5E0
 LOG 10000
CALCUL TINI 0. TEND 10.E0 NMAX 10000000
fin

*===============================================================
PLAY
CAME   1 EYE   5.00000E-01  2.21000E+00  1.21286E+01
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
FREQ 2
GO
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB PARE
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB NASN PASN
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB NORM
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
FREQ 30
GO
ENDPLAY
*===============================================================
SUIT
Post-treatment (time curves from alice file)
ECHO
RESU ALIC GARD PSCR
SORT GRAP
AXTE 1.0 'Time [s]'
COUR  1 'dy_4'  DEPL COMP 2 NOEU LECT  4 TERM
COUR  2 'dy_5'  DEPL COMP 2 NOEU LECT  5 TERM
COUR 11 'fy_4'  FEXT COMP 2 NOEU LECT  4 TERM
COUR 12 'fy_5'  FEXT COMP 2 NOEU LECT  5 TERM
TRAC 1 2 AXES 1.0 'DISPL. [M]' YZER
TRAC 11 12 AXES 1.0 'FEXT [N]' YZER
LIST 1 2 AXES 1.0 'DISPL. [M]' YZER
LIST 11 12 AXES 1.0 'FEXT [N]' YZER
*===============================================================
FIN
```

dro202a.epx

```
DRO202 Post-treatment (animation from alice file)
ECHO
RESU ALIC 'dro202.ali' GARD PSCR
SORT VISU NSTO 1
*===============================================================
PLAY
CAME   1 EYE   3.00000E+00  3.20711E+00  2.90257E+01
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
SCEN GEOM NAVI FREE
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH AVI NOCL NFTO 1001 FPS 25 KFRE 10 COMP -1 REND
FREQ 1
GOTR LOOP 999 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDPLAY
*===============================================================
FIN
```

dro203.dgibi

```
opti echo 1;
opti dime 2 elem qua4;
opti sauv form 'dro203.msh';
opti trac psc ftra 'dro203_mesh.ps';
p1 = -1 0;
p2 = 0 0;
p3 = 6 0;
p4 = 7 0;
p5 = -1 -1;
p6 = 0 6;
p7 = 6 6;
p8 = 7 -1;
p9 = -1 6;
p10 = 7 6;
n1 = 2;
n2 = 12;
tol = 1.E-3;
c1 = p1 d n1 p2;
box1 = c1 tran n2 p6;
box2 = box1 plus (7 0);
c2 = p1 d n1 p5;
box3 = c2 tran (n1 + n1 + n2) (8 0);
box = box1 et box2 et box3;
elim tol (box et p1 et p2 et p3 et p4 et p5 et p6 et p7
          et p8 et p9 et p10);
c3 = (p2 plus (0 0)) d (n1 + n1) (2 0);
squa1 = c3 tran (n1 + n1) (0 2);
depl squa1 plus (1 5);
depl squa1 tour 45 (bary squa1);
c4 = (p2 plus (0 0)) d (n1 + n1 + n1 + n1) (4 0);
rect1 = c4 tran n1 (0 1);
depl rect1 plus (1.5 3);
depl rect1 tour 30 (bary rect1);
squa2 = squa1 plus (2.5 (0 - 4.6));
depl squa2 tour (0-30) (bary squa2);
mesh = box et squa1 et rect1 et squa2;
lint = p6 d n2 p2 d n2 p3 d n2 p7;
elim tol (lint et box);
lext = p9 d n2 p1 d n1 p5 d (n1 + n2 + n1) p8 d n1 p4 d n2 p10;
elim tol (lext et box);
pbox = box elem appu larg lint;
psqua1 = squa1 elem appu larg (cont squa1);
prect1 = rect1 elem appu larg (cont rect1);
psqua2 = squa2 elem appu larg (cont squa2);
tass mesh;
sauv form mesh;
trac qual mesh;
trac qual (pbox et psqua1 et prect1 et psqua2);
fin;
```

dro203.epx

```
DRO203
ECHO
!CONV win
CAST mesh
CPLA LAGR
GEOM CAR1 mesh TERM
COMP COUL VERT LECT tous TERM
MATE LINE RO 8000.0 YOUN 1.E12 NU 0.0
          LECT tous TERM
OPTI PINS STAT
          ASN
          NORB
LINK COUP BLOQ 12 LECT lext TERM
LINK DECO PINB PENA SFAC 1.0
               BODY MLEV 4 LECT pbox   TERM
               BODY MLEV 4 LECT psqua1 TERM
               BODY MLEV 4 LECT prect1 TERM
               BODY MLEV 4 LECT psqua2 TERM
CHAR CONS GRAV 0 -9.80665D0 LECT squa1 rect1 squa2 TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA CONT ECRO TFRE 1
     FICH ALIC TFRE 1.E-2
OPTI NOTE
 CSTA 0.5E0
 LOG 10000
```

```
CALCUL TINI 0. TEND 10.E0 NMAX 1000000
fin

*=================================================================
PLAY
CAME   1 EYE   5.00000E-01  2.21000E+00  1.21286E+01
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
FREQ 2
GO
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB PARE
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB NASN PASN
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB NORM
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
FREQ 30
GO
ENDPLAY
*=================================================================
SUIT
Post-treatment (time curves from alice file)
ECHO
RESU ALIC GARD PSCR
SORT GRAP
AXTE 1.0 'Time [s]'
COUR  1 'dy_4'  DEPL COMP 2 NOEU LECT  4 TERM
COUR  2 'dy_5'  DEPL COMP 2 NOEU LECT  5 TERM
COUR 11 'fy_4'  FEXT COMP 2 NOEU LECT  4 TERM
COUR 12 'fy_5'  FEXT COMP 2 NOEU LECT  5 TERM
TRAC 1 2 AXES 1.0 'DISPL. [M]' YZER
TRAC 11 12 AXES 1.0 'FEXT [N]' YZER
LIST 1 2 AXES 1.0 'DISPL. [M]' YZER
LIST 11 12 AXES 1.0 'FEXT [N]' YZER
*=================================================================
FIN
```

dro203a.epx

```
DRO203 Post-treatment (animation from alice file)
ECHO
RESU ALIC 'dro203.ali' GARD PSCR
SORT VISU NSTO 1
*=================================================================
PLAY
CAME   1 EYE   3.00000E+00  3.20711E+00  2.90257E+01
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
SCEN GEOM NAVI FREE
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH AVI NOCL NFTO 1001 FPS 25 KFRE 10 COMP -1 REND
FREQ 1
GOTR LOOP 999 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDPLAY
*=================================================================
FIN
```

dro204.dgibi

```
opti echo 1;
opti dime 2 elem qua4;
opti sauv form 'dro204.msh';
opti trac psc ftra 'dro204_mesh.ps';
p1 = -1 0;
p2 = 0 0;
p3 = 6 0;
p4 = 7 0;
p5 = -1 -1;
p6 = 0 6;
p7 = 6 6;
p8 = 7 -1;
p9 = -1 6;
p10 = 7 6;
n1 = 2;
n2 = 12;
tol = 1.E-3;
c1 = p1 d n1 p2;
box1 = c1 tran n2 p6;
box2 = box1 plus (7 0);
c2 = p1 d n1 p5;
box3 = c2 tran (n1 + n1 + n2) (8 0);
box = box1 et box2 et box3;
elim tol (box et p1 et p2 et p3 et p4 et p5 et p6 et p7
          et p8 et p9 et p10);
c3 = (p2 plus (0 0)) d (n1 + n1) (2 0);
squa1 = c3 tran (n1 + n1) (0 2);
depl squa1 plus (1 5);
depl squa1 tour 45 (bary squa1);
c4 = (p2 plus (0 0)) d (n1 + n1 + n1 + n1) (4 0);
rect1 = c4 tran n1 (0 1);
depl rect1 plus (1.5 3);
depl rect1 tour 30 (bary rect1);
squa2 = squa1 plus (2.5 (0 - 4.6));
depl squa2 tour (0-30) (bary squa2);
mesh = box et squa1 et rect1 et squa2;
lint = p6 d n2 p2 d n2 p3 d n2 p7;
elim tol (lint et box);
lext = p9 d n2 p1 d n1 p5 d (n1 + n2 + n1) p8 d n1 p4 d n2 p10;
elim tol (lext et box);
pbox = box elem appu larg lint;
psqua1 = squa1 elem appu larg (cont squa1);
prect1 = rect1 elem appu larg (cont rect1);
psqua2 = squa2 elem appu larg (cont squa2);
tass mesh;
sauv form mesh;
trac qual mesh;
trac qual (pbox et psqua1 et prect1 et psqua2);
fin;
```

dro204.epx

```
DRO204
ECHO
!CONV win
CAST mesh
CPLA LAGR
GEOM CAR1 mesh TERM
COMP COUL VERT LECT tous TERM
MATE LINE RO 8000.0 YOUN 1.E12 NU 0.0
          LECT tous TERM
OPTI PINS STAT
         !ASN
LINK COUP BLOQ 12 LECT lext TERM
          PINB BODY MLEV 2 LECT pbox   TERM
               BODY MLEV 2 LECT psqua1 TERM
               BODY MLEV 2 LECT prect1 TERM
               BODY MLEV 2 LECT psqua2 TERM
CHAR CONS GRAV 0 -9.80665D0 LECT squa1 rect1 squa2 TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA CONT ECRO TFRE 1
     FICH ALIC TFRE 1.E-2
OPTI NOTE
 CSTA 0.5E0
 LOG 10000
CALCUL TINI 0. TEND 10.E0 NMAX 1000000
fin

*=================================================================
PLAY
CAME   1 EYE   5.00000E-01  2.21000E+00  1.21286E+01
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
FREQ 2
GO
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB PARE
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB NASN PASN
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB NORM
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
FREQ 30
GO
ENDPLAY
*=================================================================
SUIT
Post-treatment (time curves from alice file)
ECHO
RESU ALIC GARD PSCR
SORT GRAP
AXTE 1.0 'Time [s]'
COUR  1 'dy_4'  DEPL COMP 2 NOEU LECT  4 TERM
COUR  2 'dy_5'  DEPL COMP 2 NOEU LECT  5 TERM
COUR 11 'fy_4'  FEXT COMP 2 NOEU LECT  4 TERM
COUR 12 'fy_5'  FEXT COMP 2 NOEU LECT  5 TERM
TRAC 1 2 AXES 1.0 'DISPL. [M]' YZER
TRAC 11 12 AXES 1.0 'FEXT [N]' YZER
LIST 1 2 AXES 1.0 'DISPL. [M]' YZER
LIST 11 12 AXES 1.0 'FEXT [N]' YZER
*=================================================================
FIN
```

dro204a.epx

```
DRO204 Post-treatment (animation from alice file)
ECHO
RESU ALIC 'dro204.ali' GARD PSCR
SORT VISU NSTO 1
*=================================================================
PLAY
CAME   1 EYE   3.00000E+00  3.20711E+00  2.90257E+01
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
SCEN GEOM NAVI FREE
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH AVI NOCL NFTO 1001 FPS 25 KFRE 10 COMP -1 REND
FREQ 1
GOTR LOOP 999 OFFS FICH AVI CONT NOCL REND
```

```
GO
TRAC OFFS FICH AVI CONT REND
ENDPLAY
*===============================================================
FIN
```

dro205.dgibi

```
opti echo 1;
opti dime 2 elem qua4;
opti sauv form 'dro205.msh';
opti trac psc ftra 'dro205_mesh.ps';
p1 = -1 0;
p2 = 0 0;
p3 = 6 0;
p4 = 7 0;
p5 = -1 -1;
p6 = 0 6;
p7 = 6 6;
p8 = 7 -1;
p9 = -1 6;
p10 = 7 6;
n1 = 2;
n2 = 12;
tol = 1.E-3;
c1 = p1 d n1 p2;
box1 = c1 tran n2 p6;
box2 = box1 plus (7 0);
c2 = p1 d n1 p5;
box3 = c2 tran (n1 + n1 + n2) (8 0);
box = box1 et box2 et box3;
elim tol (box et p1 et p2 et p3 et p4 et p5 et p6 et p7
          et p8 et p9 et p10);
c3 = (p2 plus (0 0)) d (n1 + n1) (2 0);
squa1 = c3 tran (n1 + n1) (0 2);
depl squa1 plus (1 5);
depl squa1 tour 45 (bary squa1);
c4 = (p2 plus (0 0)) d (n1 + n1 + n1 + n1) (4 0);
rect1 = c4 tran n1 (0 1);
depl rect1 plus (1.5 3);
depl rect1 tour 30 (bary rect1);
squa2 = squa1 plus (2.5 (0 - 4.6));
depl squa2 tour (0-30) (bary squa2);
mesh = box et squa1 et rect1 et squa2;
lint = p6 d n2 p2 d n2 p3 d n2 p7;
elim tol (lint et box);
lext = p9 d n2 p1 d n1 p5 d (n1 + n2 + n1) p8 d n1 p4 d n2 p10;
elim tol (lext et box);
pbox = box elem appu larg lint;
psqua1 = squa1 elem appu larg (cont squa1);
prect1 = rect1 elem appu larg (cont rect1);
psqua2 = squa2 elem appu larg (cont squa2);
tass mesh;
sauv form mesh;
trac qual mesh;
trac qual (pbox et psqua1 et prect1 et psqua2);
fin;
```

dro205.epx

```
DRO205
ECHO
!CONV win
CAST mesh
CPLA LAGR
GEOM CAR1 mesh TERM
COMP COUL VERT LECT tous TERM
MATE LINE RO 8000.0 YOUN 1.E12 NU 0.0
          LECT tous TERM
OPTI PINS STAT
          ASN
LINK COUP BLOQ 12 LECT lext TERM
          PINB BODY MLEV 2 LECT pbox   TERM
               BODY MLEV 2 LECT psqua1 TERM
               BODY MLEV 2 LECT prect1 TERM
               BODY MLEV 2 LECT psqua2 TERM
CHAR CONS GRAV 0 -9.80665D0 LECT squa1 rect1 squa2 TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA CONT ECRO TFRE 1
     FICH ALIC TFRE 1.E-2
OPTI NOTE
 CSTA 0.5E0
 LOG 10000
CALCUL TINI 0. TEND 10.E0 NMAX 1000000
fin

*===============================================================
PLAY
CAME   1 EYE   5.00000E-01  2.21000E+00  1.21286E+01
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
FREQ 2
GO
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB PARE
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB NASN PASN
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB NORM
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
FREQ 30
GO
ENDPLAY
*===============================================================
SUIT
Post-treatment (time curves from alice file)
ECHO
RESU ALIC GARD PSCR
SORT GRAP
AXTE 1.0 'Time [s]'
COUR  1 'dy_4'  DEPL COMP 2 NOEU LECT  4 TERM
COUR  2 'dy_5'  DEPL COMP 2 NOEU LECT  5 TERM
COUR 11 'fy_4'  FEXT COMP 2 NOEU LECT  4 TERM
COUR 12 'fy_5'  FEXT COMP 2 NOEU LECT  5 TERM
TRAC 1 2 AXES 1.0 'DISPL. [M]' YZER
TRAC 11 12 AXES 1.0 'FEXT [N]' YZER
LIST 1 2 AXES 1.0 'DISPL. [M]' YZER
LIST 11 12 AXES 1.0 'FEXT [N]' YZER
*===============================================================
FIN
```

dro205a.epx

```
DRO205 Post-treatment (animation from alice file)
ECHO
RESU ALIC 'dro205.ali' GARD PSCR
SORT VISU NSTO 1
*===============================================================
PLAY
CAME   1 EYE   3.00000E+00  3.20711E+00  2.90257E+01
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
SCEN GEOM NAVI FREE
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH AVI NOCL NFTO 1001 FPS 25 KFRE 10 COMP -1 REND
FREQ 1
GOTR LOOP 999 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDPLAY
*===============================================================
FIN
```

dro206.dgibi

```
opti echo 1;
opti dime 2 elem qua4;
opti sauv form 'dro206.msh';
opti trac psc ftra 'dro206_mesh.ps';
p1 = -1 0;
p2 = 0 0;
p3 = 6 0;
p4 = 7 0;
p5 = -1 -1;
p6 = 0 6;
p7 = 6 6;
p8 = 7 -1;
p9 = -1 6;
p10 = 7 6;
n1 = 2;
n2 = 12;
tol = 1.E-3;
c1 = p1 d n1 p2;
box1 = c1 tran n2 p6;
box2 = box1 plus (7 0);
c2 = p1 d n1 p5;
box3 = c2 tran (n1 + n1 + n2) (8 0);
box = box1 et box2 et box3;
elim tol (box et p1 et p2 et p3 et p4 et p5 et p6 et p7
          et p8 et p9 et p10);
c3 = (p2 plus (0 0)) d (n1 + n1) (2 0);
squa1 = c3 tran (n1 + n1) (0 2);
depl squa1 plus (1 5);
depl squa1 tour 45 (bary squa1);
c4 = (p2 plus (0 0)) d (n1 + n1 + n1 + n1) (4 0);
rect1 = c4 tran n1 (0 1);
depl rect1 plus (1.5 3);
depl rect1 tour 30 (bary rect1);
squa2 = squa1 plus (2.5 (0 - 4.6));
depl squa2 tour (0-30) (bary squa2);
mesh = box et squa1 et rect1 et squa2;
lint = p6 d n2 p2 d n2 p3 d n2 p7;
elim tol (lint et box);
lext = p9 d n2 p1 d n1 p5 d (n1 + n2 + n1) p8 d n1 p4 d n2 p10;
elim tol (lext et box);
pbox = box elem appu larg lint;
psqua1 = squa1 elem appu larg (cont squa1);
prect1 = rect1 elem appu larg (cont rect1);
psqua2 = squa2 elem appu larg (cont squa2);
tass mesh;
sauv form mesh;
trac qual mesh;
trac qual (pbox et psqua1 et prect1 et psqua2);
fin;
```

dro206.epx

```
DRO206
ECHO
!CONV win
CAST mesh
CPLA LAGR
GEOM CAR1 mesh TERM
COMP COUL VERT LECT tous TERM
MATE LINE RO 8000.0 YOUN 1.E12 NU 0.0
          LECT tous TERM
OPTI PINS STAT
```

```
        !ASN
LINK COUP BLOQ 12 LECT lext TERM
          PINB BODY MLEV 4 LECT pbox   TERM
               BODY MLEV 4 LECT psqua1 TERM
               BODY MLEV 4 LECT prect1 TERM
               BODY MLEV 4 LECT psqua2 TERM
CHAR CONS GRAV 0 -9.80665D0 LECT squa1 rect1 squa2 TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA CONT ECRO TFRE 1
     FICH ALIC TFRE 1.E-2
OPTI NOTE
 CSTA 0.5E0
 LOG 10000
CALCUL TINI 0. TEND 10.E0 NMAX 1000000
fin

*=================================================================
PLAY
CAME   1 EYE   5.00000E-01  2.21000E+00  1.21286E+01
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
FREQ 2
GO
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB PARE
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB NASN PASN
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB NORM
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
FREQ 30
GO
ENDPLAY
*=================================================================
SUIT
Post-treatment (time curves from alice file)
ECHO
RESU ALIC GARD PSCR
SORT GRAP
AXTE 1.0 'Time [s]'
COUR  1 'dy_4'  DEPL COMP 2 NOEU LECT  4 TERM
COUR  2 'dy_5'  DEPL COMP 2 NOEU LECT  5 TERM
COUR 11 'fy_4'  FEXT COMP 2 NOEU LECT  4 TERM
COUR 12 'fy_5'  FEXT COMP 2 NOEU LECT  5 TERM
TRAC 1 2 AXES 1.0 'DISPL. [M]' YZER
TRAC 11 12 AXES 1.0 'FEXT [N]' YZER
LIST 1 2 AXES 1.0 'DISPL. [M]' YZER
LIST 11 12 AXES 1.0 'FEXT [N]' YZER
*=================================================================
FIN
```

dro206a.epx

```
DRO206 Post-treatment (animation from alice file)
ECHO
RESU ALIC 'dro206.ali' GARD PSCR
SORT VISU NSTO 1
*=================================================================
PLAY
CAME   1 EYE   3.00000E+00  3.20711E+00  2.90257E+01
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
SCEN GEOM NAVI FREE
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH AVI NOCL NFTO 1001 FPS 25 KFRE 10 COMP -1 REND
FREQ 1
GOTR LOOP 999 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDPLAY
*=================================================================
FIN
```

dro207.dgibi

```
opti echo 1;
opti dime 2 elem qua4;
opti sauv form 'dro207.msh';
opti trac psc ftra 'dro207_mesh.ps';
p1 = -1 0;
p2 = 0 0;
p3 = 6 0;
p4 = 7 0;
p5 = -1 -1;
p6 = 0 6;
p7 = 6 6;
p8 = 7 -1;
p9 = -1 6;
p10 = 7 6;
n1 = 2;
n2 = 12;
tol = 1.E-3;
c1 = p1 d n1 p2;
box1 = c1 tran n2 p6;
box2 = box1 plus (7 0);
c2 = p1 d n1 p5;
box3 = c2 tran (n1 + n1 + n2) (8 0);
box = box1 et box2 et box3;
elim tol (box et p1 et p2 et p3 et p4 et p5 et p6 et p7
          et p8 et p9 et p10);
c3 = (p2 plus (0 0)) d (n1 + n1) (2 0);
squa1 = c3 tran (n1 + n1) (0 2);
depl squa1 plus (1 5);
depl squa1 tour 45 (bary squa1);
c4 = (p2 plus (0 0)) d (n1 + n1 + n1 + n1) (4 0);
rect1 = c4 tran n1 (0 1);
depl rect1 plus (1.5 3);
depl rect1 tour 30 (bary rect1);
squa2 = squa1 plus (2.5 (0 - 4.6));
depl squa2 tour (0-30) (bary squa2);
mesh = box et squa1 et rect1 et squa2;
lint = p6 d n2 p2 d n2 p3 d n2 p7;
elim tol (lint et box);
lext = p9 d n2 p1 d n1 p5 d (n1 + n2 + n1) p8 d n1 p4 d n2 p10;
elim tol (lext et box);
pbox = box elem appu larg lint;
psqua1 = squa1 elem appu larg (cont squa1);
prect1 = rect1 elem appu larg (cont rect1);
psqua2 = squa2 elem appu larg (cont squa2);
tass mesh;
sauv form mesh;
trac qual mesh;
trac qual (pbox et psqua1 et prect1 et psqua2);
fin;
```

dro207.epx

```
DRO207
ECHO
!CONV win
CAST mesh
CPLA LAGR
GEOM CAR1 mesh TERM
COMP COUL VERT LECT tous TERM
MATE LINE RO 8000.0 YOUN 1.E12 NU 0.0
         LECT tous TERM
OPTI PINS STAT
         ASN
         NORB
LINK COUP BLOQ 12 LECT lext TERM
LINK DECO PINB PENA SFAC 0.1
              BODY MLEV 2 LECT pbox   TERM
              BODY MLEV 2 LECT psqua1 TERM
              BODY MLEV 2 LECT prect1 TERM
              BODY MLEV 2 LECT psqua2 TERM
CHAR CONS GRAV 0 -9.80665D0 LECT squa1 rect1 squa2 TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA CONT ECRO TFRE 1
     FICH ALIC TFRE 1.E-2
OPTI NOTE
 CSTA 0.5E0
 LOG 10000
CALCUL TINI 0. TEND 10.E0 NMAX 1000000
fin

*=================================================================
PLAY
CAME   1 EYE   5.00000E-01  2.21000E+00  1.21286E+01
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
FREQ 2
GO
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB PARE
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB NASN PASN
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB NORM
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
FREQ 30
GO
ENDPLAY
*=================================================================
SUIT
Post-treatment (time curves from alice file)
ECHO
RESU ALIC GARD PSCR
SORT GRAP
AXTE 1.0 'Time [s]'
COUR  1 'dy_4'  DEPL COMP 2 NOEU LECT  4 TERM
COUR  2 'dy_5'  DEPL COMP 2 NOEU LECT  5 TERM
COUR 11 'fy_4'  FEXT COMP 2 NOEU LECT  4 TERM
COUR 12 'fy_5'  FEXT COMP 2 NOEU LECT  5 TERM
TRAC 1 2 AXES 1.0 'DISPL. [M]' YZER
TRAC 11 12 AXES 1.0 'FEXT [N]' YZER
LIST 1 2 AXES 1.0 'DISPL. [M]' YZER
LIST 11 12 AXES 1.0 'FEXT [N]' YZER
*=================================================================
FIN
```

dro207a.epx

```
DRO207 Post-treatment (animation from alice file)
ECHO
RESU ALIC 'dro207.ali' GARD PSCR
SORT VISU NSTO 1
```

```
*================================================================
PLAY
CAME   1 EYE   3.00000E+00  3.20711E+00  2.90257E+01
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
SCEN GEOM NAVI FREE
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH AVI NOCL NFTO 1001 FPS 25 KFRE 10 COMP -1 REND
FREQ 1
GOTR LOOP 999 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDPLAY
*================================================================
FIN
```

dro208.dgibi

```
opti echo 1;
opti dime 2 elem qua4;
opti sauv form 'dro208.msh';
opti trac psc ftra 'dro208_mesh.ps';
p1 = -1 0;
p2 = 0 0;
p3 = 6 0;
p4 = 7 0;
p5 = -1 -1;
p6 = 0 6;
p7 = 6 6;
p8 = 7 -1;
p9 = -1 6;
p10 = 7 6;
n1 = 2;
n2 = 12;
tol = 1.E-3;
c1 = p1 d n1 p2;
box1 = c1 tran n2 p6;
box2 = box1 plus (7 0);
c2 = p1 d n1 p5;
box3 = c2 tran (n1 + n1 + n2) (8 0);
box = box1 et box2 et box3;
elim tol (box et p1 et p2 et p3 et p4 et p5 et p6 et p7
          et p8 et p9 et p10);
c3 = (p2 plus (0 0)) d (n1 + n1) (2 0);
squa1 = c3 tran (n1 + n1) (0 2);
depl squa1 plus (1 5);
depl squa1 tour 45 (bary squa1);
c4 = (p2 plus (0 0)) d (n1 + n1 + n1 + n1) (4 0);
rect1 = c4 tran n1 (0 1);
depl rect1 plus (1.5 3);
depl rect1 tour 30 (bary rect1);
squa2 = squa1 plus (2.5 (0 - 4.6));
depl squa2 tour (0-30) (bary squa2);
mesh = box et squa1 et rect1 et squa2;
lint = p6 d n2 p2 d n2 p3 d n2 p7;
elim tol (lint et box);
lext = p9 d n2 p1 d n1 p5 d (n1 + n2 + n1) p8 d n1 p4 d n2 p10;
elim tol (lext et box);
pbox = box elem appu larg lint;
psqua1 = squa1 elem appu larg (cont squa1);
prect1 = rect1 elem appu larg (cont rect1);
psqua2 = squa2 elem appu larg (cont squa2);
tass mesh;
sauv form mesh;
trac qual mesh;
trac qual (pbox et psqua1 et prect1 et psqua2);
fin;
```

dro208.epx

```
DRO208
ECHO
!CONV win
CAST mesh
CPLA LAGR
GEOM CAR1 mesh TERM
COMP COUL VERT LECT tous TERM
MATE LINE RO 8000.0 YOUN 1.E12 NU 0.0
          LECT tous TERM
OPTI PINS STAT
          ASN
          NORB
LINK COUP BLOQ 12 LECT lext TERM
LINK DECO PINB PENA SFAC 10.0
               BODY MLEV 2 LECT pbox   TERM
               BODY MLEV 2 LECT psqua1 TERM
               BODY MLEV 2 LECT prect1 TERM
               BODY MLEV 2 LECT psqua2 TERM
CHAR CONS GRAV 0 -9.80665D0 LECT squa1 rect1 squa2 TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA CONT ECRO TFRE 1
     FICH ALIC TFRE 1.E-2
OPTI NOTE
 CSTA 0.5E0
 LOG 10000
CALCUL TINI 0. TEND 10.E0 NMAX 1000000
fin

*================================================================
PLAY
CAME   1 EYE   5.00000E-01  2.21000E+00  1.21286E+01
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
FREQ 2
GO
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB PARE
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB NASN PASN
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB NORM
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
FREQ 30
GO
ENDPLAY
*================================================================
SUIT
Post-treatment (time curves from alice file)
ECHO
RESU ALIC GARD PSCR
SORT GRAP
AXTE 1.0 'Time [s]'
COUR  1 'dy_4'  DEPL COMP 2 NOEU LECT  4 TERM
COUR  2 'dy_5'  DEPL COMP 2 NOEU LECT  5 TERM
COUR 11 'fy_4'  FEXT COMP 2 NOEU LECT  4 TERM
COUR 12 'fy_5'  FEXT COMP 2 NOEU LECT  5 TERM
TRAC 1 2 AXES 1.0 'DISPL. [M]' YZER
TRAC 11 12 AXES 1.0 'FEXT [N]' YZER
LIST 1 2 AXES 1.0 'DISPL. [M]' YZER
LIST 11 12 AXES 1.0 'FEXT [N]' YZER
*================================================================
FIN
```

dro208a.epx

```
DRO208 Post-treatment (animation from alice file)
ECHO
RESU ALIC 'dro208.ali' GARD PSCR
SORT VISU NSTO 1
*================================================================
PLAY
CAME   1 EYE   3.00000E+00  3.20711E+00  2.90257E+01
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
SCEN GEOM NAVI FREE
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH AVI NOCL NFTO 1001 FPS 25 KFRE 10 COMP -1 REND
FREQ 1
GOTR LOOP 999 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDPLAY
*================================================================
FIN
```

dro209.dgibi

```
opti echo 1;
opti dime 2 elem qua4;
opti sauv form 'dro209.msh';
opti trac psc ftra 'dro209_mesh.ps';
p1 = -1 0;
p2 = 0 0;
p3 = 6 0;
p4 = 7 0;
p5 = -1 -1;
p6 = 0 6;
p7 = 6 6;
p8 = 7 -1;
p9 = -1 6;
p10 = 7 6;
n1 = 2;
n2 = 12;
tol = 1.E-3;
c1 = p1 d n1 p2;
box1 = c1 tran n2 p6;
box2 = box1 plus (7 0);
c2 = p1 d n1 p5;
box3 = c2 tran (n1 + n1 + n2) (8 0);
box = box1 et box2 et box3;
elim tol (box et p1 et p2 et p3 et p4 et p5 et p6 et p7
          et p8 et p9 et p10);
c3 = (p2 plus (0 0)) d (n1 + n1) (2 0);
squa1 = c3 tran (n1 + n1) (0 2);
depl squa1 plus (1 5);
depl squa1 tour 45 (bary squa1);
c4 = (p2 plus (0 0)) d (n1 + n1 + n1 + n1) (4 0);
rect1 = c4 tran n1 (0 1);
depl rect1 plus (1.5 3);
depl rect1 tour 30 (bary rect1);
squa2 = squa1 plus (2.5 (0 - 4.6));
depl squa2 tour (0-30) (bary squa2);
mesh = box et squa1 et rect1 et squa2;
lint = p6 d n2 p2 d n2 p3 d n2 p7;
elim tol (lint et box);
lext = p9 d n2 p1 d n1 p5 d (n1 + n2 + n1) p8 d n1 p4 d n2 p10;
elim tol (lext et box);
pbox = box elem appu larg lint;
psqua1 = squa1 elem appu larg (cont squa1);
prect1 = rect1 elem appu larg (cont rect1);
psqua2 = squa2 elem appu larg (cont squa2);
tass mesh;
sauv form mesh;
trac qual mesh;
trac qual (pbox et psqua1 et prect1 et psqua2);
```

```
fin;
```

dro209.epx

```
DRO209
ECHO
!CONV win
CAST mesh
CPLA LAGR
GEOM CAR1 mesh TERM
COMP COUL VERT LECT tous TERM
MATE LINE RO 8000.0 YOUN 1.E12 NU 0.0
          LECT tous TERM
OPTI PINS STAT
          ASN
          NORB
LINK COUP BLOQ 12 LECT lext TERM
LINK DECO PINB PENA SFAC 10.0
               BODY MLEV 4 LECT pbox   TERM
               BODY MLEV 4 LECT psqua1 TERM
               BODY MLEV 4 LECT prect1 TERM
               BODY MLEV 4 LECT psqua2 TERM
CHAR CONS GRAV 0 -9.80665D0 LECT squa1 rect1 squa2 TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA CONT ECRO TFRE 1
     FICH ALIC TFRE 1.E-2
OPTI NOTE
 CSTA 0.5E0
 LOG 10000
CALCUL TINI 0. TEND 10.E0 NMAX 1000000
fin

*=================================================================
PLAY
CAME   1 EYE   5.00000E-01  2.21000E+00  1.21286E+01
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
FREQ 2
GO
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB PARE
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB NASN PASN
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB NORM
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
FREQ 30
GO
ENDPLAY
*=================================================================
SUIT
Post-treatment (time curves from alice file)
ECHO
RESU ALIC GARD PSCR
SORT GRAP
AXTE 1.0 'Time [s]'
COUR  1 'dy_4'  DEPL COMP 2 NOEU LECT  4 TERM
COUR  2 'dy_5'  DEPL COMP 2 NOEU LECT  5 TERM
COUR 11 'fy_4'  FEXT COMP 2 NOEU LECT  4 TERM
COUR 12 'fy_5'  FEXT COMP 2 NOEU LECT  5 TERM
TRAC 1 2 AXES 1.0 'DISPL. [M]' YZER
TRAC 11 12 AXES 1.0 'FEXT [N]' YZER
LIST 1 2 AXES 1.0 'DISPL. [M]' YZER
LIST 11 12 AXES 1.0 'FEXT [N]' YZER
*=================================================================
FIN
```

dro209a.epx

```
DRO209 Post-treatment (animation from alice file)
ECHO
RESU ALIC 'dro209.ali' GARD PSCR
SORT VISU NSTO 1
*=================================================================
PLAY
CAME   1 EYE   3.00000E+00  3.20711E+00  2.90257E+01
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
SCEN GEOM NAVI FREE
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH AVI NOCL NFTO 1001 FPS 25 KFRE 10 COMP -1 REND
FREQ 1
GOTR LOOP 999 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDPLAY
*=================================================================
FIN
```

dro210.dgibi

```
opti echo 1;
opti dime 2 elem qua4;
opti sauv form 'dro210.msh';
opti trac psc ftra 'dro210_mesh.ps';
p1 = -1 0;
p2 = 0 0;
p3 = 6 0;
p4 = 7 0;
p5 = -1 -1;
p6 = 0 6;
p7 = 6 6;
p8 = 7 -1;
p9 = -1 6;
p10 = 7 6;
n1 = 2;
n2 = 12;
tol = 1.E-3;
c1 = p1 d n1 p2;
box1 = c1 tran n2 p6;
box2 = box1 plus (7 0);
c2 = p1 d n1 p5;
box3 = c2 tran (n1 + n1 + n2) (8 0);
box = box1 et box2 et box3;
elim tol (box et p1 et p2 et p3 et p4 et p5 et p6 et p7
          et p8 et p9 et p10);
c3 = (p2 plus (0 0)) d (n1 + n1) (2 0);
squa1 = c3 tran (n1 + n1) (0 2);
depl squa1 plus (1 5);
depl squa1 tour 45 (bary squa1);
c4 = (p2 plus (0 0)) d (n1 + n1 + n1 + n1) (4 0);
rect1 = c4 tran n1 (0 1);
depl rect1 plus (1.5 3);
depl rect1 tour 30 (bary rect1);
squa2 = squa1 plus (2.5 (0 - 4.6));
depl squa2 tour (0-30) (bary squa2);
mesh = box et squa1 et rect1 et squa2;
lint = p6 d n2 p2 d n2 p3 d n2 p7;
elim tol (lint et box);
lext = p9 d n2 p1 d n1 p5 d (n1 + n2 + n1) p8 d n1 p4 d n2 p10;
elim tol (lext et box);
pbox = box elem appu larg lint;
psqua1 = squa1 elem appu larg (cont squa1);
prect1 = rect1 elem appu larg (cont rect1);
psqua2 = squa2 elem appu larg (cont squa2);
tass mesh;
sauv form mesh;
trac qual mesh;
trac qual (pbox et psqua1 et prect1 et psqua2);
fin;
```

dro210.epx

```
DRO210
ECHO
!CONV win
CAST mesh
CPLA LAGR
GEOM CAR1 mesh TERM
COMP COUL VERT LECT tous TERM
MATE LINE RO 8000.0 YOUN 1.E12 NU 0.0
          LECT tous TERM
OPTI PINS STAT
          ASN
          NORB
LINK COUP BLOQ 12 LECT lext TERM
LINK DECO PINB PENA SFAC 100.0
               BODY MLEV 2 LECT pbox   TERM
               BODY MLEV 2 LECT psqua1 TERM
               BODY MLEV 2 LECT prect1 TERM
               BODY MLEV 2 LECT psqua2 TERM
CHAR CONS GRAV 0 -9.80665D0 LECT squa1 rect1 squa2 TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA CONT ECRO TFRE 1
     FICH ALIC TFRE 1.E-2
OPTI NOTE
 CSTA 0.5E0
 LOG 10000
CALCUL TINI 0. TEND 10.E0 NMAX 1000000
fin

*=================================================================
PLAY
CAME   1 EYE   5.00000E-01  2.21000E+00  1.21286E+01
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
FREQ 2
GO
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB PARE
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB NASN PASN
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB NORM
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
FREQ 30
GO
ENDPLAY
*=================================================================
SUIT
Post-treatment (time curves from alice file)
ECHO
RESU ALIC GARD PSCR
SORT GRAP
AXTE 1.0 'Time [s]'
COUR  1 'dy_4'  DEPL COMP 2 NOEU LECT  4 TERM
COUR  2 'dy_5'  DEPL COMP 2 NOEU LECT  5 TERM
```

```
COUR 11 'fy_4'  FEXT COMP 2 NOEU LECT  4 TERM
COUR 12 'fy_5'  FEXT COMP 2 NOEU LECT  5 TERM
TRAC 1 2 AXES 1.0 'DISPL. [M]' YZER
TRAC 11 12 AXES 1.0 'FEXT [N]' YZER
LIST 1 2 AXES 1.0 'DISPL. [M]' YZER
LIST 11 12 AXES 1.0 'FEXT [N]' YZER
*================================================================
FIN
```

dro210a.epx

```
DRO210 Post-treatment (animation from alice file)
ECHO
RESU ALIC 'dro210.ali' GARD PSCR
SORT VISU NSTO 1
*================================================================
PLAY
CAME   1 EYE   3.00000E+00  3.20711E+00  2.90257E+01
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
SCEN GEOM NAVI FREE
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH AVI NOCL NFTO 799 FPS 25 KFRE 10 COMP -1 REND
FREQ 1
GOTR LOOP 797 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDPLAY
*================================================================
FIN
```

dro211.dgibi

```
opti echo 1;
opti dime 2 elem qua4;
opti sauv form 'dro211.msh';
opti trac psc ftra 'dro211_mesh.ps';
p1 = -1 0;
p2 = 0 0;
p3 = 6 0;
p4 = 7 0;
p5 = -1 -1;
p6 = 0 6;
p7 = 6 6;
p8 = 7 -1;
p9 = -1 6;
p10 = 7 6;
n1 = 2;
n2 = 12;
tol = 1.E-3;
c1 = p1 d n1 p2;
box1 = c1 tran n2 p6;
box2 = box1 plus (7 0);
c2 = p1 d n1 p5;
box3 = c2 tran (n1 + n1 + n2) (8 0);
box = box1 et box2 et box3;
elim tol (box et p1 et p2 et p3 et p4 et p5 et p6 et p7
          et p8 et p9 et p10);
c3 = (p2 plus (0 0)) d (n1 + n1) (2 0);
squa1 = c3 tran (n1 + n1) (0 2);
depl squa1 plus (1 5);
depl squa1 tour 45 (bary squa1);
c4 = (p2 plus (0 0)) d (n1 + n1 + n1 + n1) (4 0);
rect1 = c4 tran n1 (0 1);
depl rect1 plus (1.5 3);
depl rect1 tour 30 (bary rect1);
squa2 = squa1 plus (2.5 (0 - 4.6));
depl squa2 tour (0-30) (bary squa2);
mesh = box et squa1 et rect1 et squa2;
lint = p6 d n2 p2 d n2 p3 d n2 p7;
elim tol (lint et box);
lext = p9 d n2 p1 d n1 p5 d (n1 + n2 + n1) p8 d n1 p4 d n2 p10;
elim tol (lext et box);
pbox = box elem appu larg lint;
psqua1 = squa1 elem appu larg (cont squa1);
prect1 = rect1 elem appu larg (cont rect1);
psqua2 = squa2 elem appu larg (cont squa2);
tass mesh;
sauv form mesh;
trac qual mesh;
trac qual (pbox et psqua1 et prect1 et psqua2);
fin;
```

dro211.epx

```
DRO211
ECHO
!CONV win
CAST mesh
CPLA LAGR
GEOM CAR1 mesh TERM
COMP COUL VERT LECT tous TERM
MATE LINE RO 8000.0 YOUN 1.E12 NU 0.0
          LECT tous TERM
OPTI PINS STAT
          ASN
          REB1
LINK COUP BLOQ 12 LECT lext TERM
LINK DECO PINB PENA SFAC 1.0
               BODY MLEV 2 LECT pbox   TERM
               BODY MLEV 2 LECT psqua1 TERM
               BODY MLEV 2 LECT prect1 TERM
               BODY MLEV 2 LECT psqua2 TERM
CHAR CONS GRAV 0 -9.80665D0 LECT squa1 rect1 squa2 TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA CONT ECRO TFRE 1
     FICH ALIC TFRE 1.E-2
OPTI NOTE
 CSTA 0.5E0
 LOG 10000
CALCUL TINI 0. TEND 10.E0 NMAX 1000000
fin

*================================================================
PLAY
CAME   1 EYE   5.00000E-01  2.21000E+00  1.21286E+01
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
FREQ 2
GO
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB PARE
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB NASN PASN
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB NORM
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
FREQ 30
GO
ENDPLAY
*================================================================
SUIT
Post-treatment (time curves from alice file)
ECHO
RESU ALIC GARD PSCR
SORT GRAP
AXTE 1.0 'Time [s]'
COUR  1 'dy_4'  DEPL COMP 2 NOEU LECT  4 TERM
COUR  2 'dy_5'  DEPL COMP 2 NOEU LECT  5 TERM
COUR 11 'fy_4'  FEXT COMP 2 NOEU LECT  4 TERM
COUR 12 'fy_5'  FEXT COMP 2 NOEU LECT  5 TERM
TRAC 1 2 AXES 1.0 'DISPL. [M]' YZER
TRAC 11 12 AXES 1.0 'FEXT [N]' YZER
LIST 1 2 AXES 1.0 'DISPL. [M]' YZER
LIST 11 12 AXES 1.0 'FEXT [N]' YZER
*================================================================
FIN
```

dro211a.epx

```
DRO211 Post-treatment (animation from alice file)
ECHO
RESU ALIC 'dro211.ali' GARD PSCR
SORT VISU NSTO 1
*================================================================
PLAY
CAME   1 EYE   3.00000E+00  3.20711E+00  2.90257E+01
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
SCEN GEOM NAVI FREE
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH AVI NOCL NFTO 1001 FPS 25 KFRE 10 COMP -1 REND
FREQ 1
GOTR LOOP 999 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDPLAY
*================================================================
FIN
```

pene01.epx

```
PENE01
ECHO
!CONV win
CPLA LAGR
GEOM LIBR POIN 8 CAR1 2 TERM
 0 0  1 0  0 1  1 1
 0 1  1 1  0 2  1 2
 1 2 4 3
 5 6 8 7
COMP COUL VERT LECT tous TERM
MATE LINE RO 8000.0 YOUN 1.E12 NU 0.0
          LECT tous TERM
INIT VITE 2  50 LECT 1 PAS 1 4 TERM
     VITE 2 -50 LECT 5 PAS 1 8 TERM
OPTI PINS DUMP
          STAT
         !VIDE
         !EQVL
         !EQVD
          ASN
         !NORB
LINK DECO PINB PENA SFAC 1.0
               BODY LECT 1 TERM
               BODY LECT 2 TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA CONT ECRO FREQ 1
     FICH ALIC FREQ 1
OPTI NOTE
 CSTA 0.5E0
 LOG 1
CALCUL TINI 0. TEND 100.E-3 NMAX 15
*================================================================
```

```
PLAY
CAME   1 EYE   5.00000E-01  1.00000E+00  7.83758E+00
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB PARE
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB NASN PASN NORM
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
ENDPLAY
*==================================================================
FIN
```

pene02.eco

```
STANDARD VERSION

 TITLE: PENE02

      3>!CONV win
      4>CPLA LAGR
      5>GEOM LIBR POIN 12 CAR1 4 TERM
 == SET FLAG IN INPUT FILE
 == GO BACK TO FLAG IN INPUT FILE
 == SET FLAG IN INPUT FILE
 == GO BACK TO FLAG IN INPUT FILE
      6> 0 0  1 0  0 1  1 1  0 2  1 2
      7> 0 2.42  1 2.42  0 3.42  1 3.42  0 4.42  1 4.42
      8> 1 2 4 3
      9> 3 4 6 5
     10> 7 8 10 9
     11> 9 10 12 11
     12>COMP COUL VERT LECT tous TERM
     13>MATE LINE RO 8000.0 YOUN 1.E12 NU 0.0
     14>          LECT tous TERM
     15>INIT VITE 2  50 LECT 1 PAS 1 6 TERM
     16>     VITE 2 -50 LECT 7 PAS 1 12 TERM
     17>OPTI PINS DUMP
     18>          STAT
     19>
     19>!VIDE
     20>
     20>!EQVL
     21>
     21>!EQVD
     22>          ASN
     23>
     23>!NORB
     24>LINK DECO PINB PENA SFAC 1.0
     25>               BODY LECT 2 TERM
     26>               BODY LECT 3 TERM
     27>ECRI COOR DEPL VITE ACCE FINT FEXT FLIA CONT ECRO FREQ 1
     28>     FICH ALIC FREQ 1
     29>OPTI NOTE
     30> CSTA 0.5E0
     31> LOG 1
     32>CALCUL TINI 0. TEND 100.E-3 NMAX 20
     33>*==================================================================
     34>PLAY

 -> END OF INITIALISATIONS ** TCPU =     0.02 SEC.

     35>CAME   1 EYE   5.00000E-01  2.21000E+00  1.21286E+01
     36>!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
     37>         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
     38>         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
     39>         UP    0.00000E+00  1.00000E+00  0.00000E+00
     40>         FOV   2.48819E+01
     41>FREQ 2
 PREVIOUS FREQUENCY (BY STEP) OF VISUALISATION:       1
 CHOOSE NEW FREQUENCY:
 NEW FREQUENCY (BY STEP) OF VISUALISATION:       2
     42>GO
  >  END OF TIME STEP NO :       0 * T = 0.00000E+00 * TCPU =      0.02 SEC.
  >  END OF TIME STEP NO :       1 * T = 3.87298E-05 * TCPU =      0.02 SEC.
     43>SCEN GEOM NAVI FREE
     44>          FACE HFRO
     45>          PINB PARE
     46>     COLO PAPE
     47>SLER CAM1 1 NFRA 1
     48>TRAC OFFS FICH BMP REND
     49>SCEN GEOM NAVI FREE
     50>          FACE HFRO
     51>          PINB NASN PASN
     52>     COLO PAPE
     53>SLER CAM1 1 NFRA 1
     54>TRAC OFFS FICH BMP REND
     55>SCEN GEOM NAVI FREE
     56>          FACE HFRO
     57>          PINB NORM
     58>     COLO PAPE
     59>SLER CAM1 1 NFRA 1
     60>TRAC OFFS FICH BMP REND
     61>FREQ 20
 PREVIOUS FREQUENCY (BY STEP) OF VISUALISATION:       2
 CHOOSE NEW FREQUENCY:
 NEW FREQUENCY (BY STEP) OF VISUALISATION:      20
     62>GO
  >  END OF TIME STEP NO :       2 * T = 7.74597E-05 * TCPU =      0.08 SEC.
  >  END OF TIME STEP NO :       3 * T = 1.16190E-04 * TCPU =      0.08 SEC.
  >  END OF TIME STEP NO :       4 * T = 1.54917E-04 * TCPU =      0.08 SEC.
  >  END OF TIME STEP NO :       5 * T = 1.93637E-04 * TCPU =      0.08 SEC.
  >  END OF TIME STEP NO :       6 * T = 2.32347E-04 * TCPU =      0.08 SEC.
  >  END OF TIME STEP NO :       7 * T = 2.71045E-04 * TCPU =      0.08 SEC.
  >  END OF TIME STEP NO :       8 * T = 3.09734E-04 * TCPU =      0.08 SEC.
  >  END OF TIME STEP NO :       9 * T = 3.48422E-04 * TCPU =      0.08 SEC.
  >  END OF TIME STEP NO :      10 * T = 3.87114E-04 * TCPU =      0.08 SEC.
  >  END OF TIME STEP NO :      11 * T = 4.25818E-04 * TCPU =      0.08 SEC.
  >  END OF TIME STEP NO :      12 * T = 4.64536E-04 * TCPU =      0.09 SEC.
  >  END OF TIME STEP NO :      13 * T = 5.03266E-04 * TCPU =      0.09 SEC.
  >  END OF TIME STEP NO :      14 * T = 5.42005E-04 * TCPU =      0.09 SEC.
  >  END OF TIME STEP NO :      15 * T = 5.80735E-04 * TCPU =      0.09 SEC.
  >  END OF TIME STEP NO :      16 * T = 6.19460E-04 * TCPU =      0.09 SEC.
  >  END OF TIME STEP NO :      17 * T = 6.58181E-04 * TCPU =      0.09 SEC.
  >  END OF TIME STEP NO :      18 * T = 6.96903E-04 * TCPU =      0.09 SEC.
  >  END OF TIME STEP NO :      19 * T = 7.35627E-04 * TCPU =      0.09 SEC.
     63>ENDPLAY
  >  END OF TIME STEP NO :      20 * T = 7.74347E-04 * TCPU =      0.09 SEC.
     64>*==================================================================
     65>SUIT
     66>Post-treatment (time curves from alice file)

 TITLE: POST-TREATMENT (TIME CURVES FROM ALICE FILE)

     67>ECHO
     68>RESU ALIC GARD PSCR
     69>SORT GRAP
EDITION OF "ALICE" FILE fort.10 (VERSION 15) WITH THE TITLE:
 PENE02
THE FILE CREATED ON: 06-06-2014
 ALICE FILE N 10 STORAGE N    1 AT STEP       0 TIME  0.00000E+00
 ALICE FILE N 10 STORAGE N    2 AT STEP       1 TIME  3.87298E-05
 ALICE FILE N 10 STORAGE N    3 AT STEP       2 TIME  7.74597E-05
 ALICE FILE N 10 STORAGE N    4 AT STEP       3 TIME  1.16189E-04
 ALICE FILE N 10 STORAGE N    5 AT STEP       4 TIME  1.54917E-04
 ALICE FILE N 10 STORAGE N    6 AT STEP       5 TIME  1.93637E-04
 ALICE FILE N 10 STORAGE N    7 AT STEP       6 TIME  2.32347E-04
 ALICE FILE N 10 STORAGE N    8 AT STEP       7 TIME  2.71045E-04
 ALICE FILE N 10 STORAGE N    9 AT STEP       8 TIME  3.09734E-04
 ALICE FILE N 10 STORAGE N   10 AT STEP       9 TIME  3.48422E-04
 ALICE FILE N 10 STORAGE N   11 AT STEP      10 TIME  3.87114E-04
 ALICE FILE N 10 STORAGE N   12 AT STEP      11 TIME  4.25818E-04
 ALICE FILE N 10 STORAGE N   13 AT STEP      12 TIME  4.64536E-04
 ALICE FILE N 10 STORAGE N   14 AT STEP      13 TIME  5.03266E-04
 ALICE FILE N 10 STORAGE N   15 AT STEP      14 TIME  5.42005E-04
 ALICE FILE N 10 STORAGE N   16 AT STEP      15 TIME  5.80736E-04
 ALICE FILE N 10 STORAGE N   17 AT STEP      16 TIME  6.19460E-04
 ALICE FILE N 10 STORAGE N   18 AT STEP      17 TIME  6.58181E-04
 ALICE FILE N 10 STORAGE N   19 AT STEP      18 TIME  6.96903E-04
 ALICE FILE N 10 STORAGE N   20 AT STEP      19 TIME  7.35627E-04
 ALICE FILE N 10 STORAGE N   21 AT STEP      20 TIME  7.74347E-04
FILE CONTAINS :      21 TIME INSTANTS.
TMIN :    0.00000E+00  TMAX :      7.74347E-04
 == SET FLAG IN INPUT FILE
 == GO BACK TO FLAG IN INPUT FILE
     70>AXTE 1.0 'Time [s]'
     71>COUR  1 'dy_5'  DEPL COMP 2 NOEU LECT  5 TERM
     72>COUR  2 'dy_7'  DEPL COMP 2 NOEU LECT  7 TERM
     73>COUR 11 'fy_5'  FEXT COMP 2 NOEU LECT  5 TERM
     74>COUR 12 'fy_7'  FEXT COMP 2 NOEU LECT  7 TERM
     75>TRAC 1 2 AXES 1.0 'DISPL. [M]' YZER
     76>TRAC 11 12 AXES 1.0 'FEXT [N]' YZER
     77>LIST 1 2 AXES 1.0 'DISPL. [M]' YZER
     78>LIST 11 12 AXES 1.0 'FEXT [N]' YZER
     79>*==================================================================
     80>FIN
   >  TOTAL CPU TIME :        0.09 SEC.
Saving image 0001 to pene02_0001.bmp ... Checking OpenGL errors ...
bitmap saved.
Saving image 0002 to pene02_0002.bmp ... Checking OpenGL errors ...
bitmap saved.
Saving image 0003 to pene02_0003.bmp ... Checking OpenGL errors ...
bitmap saved.
```

pene02.epx

```
PENE02
ECHO
!CONV win
CPLA LAGR
GEOM LIBR POIN 12 CAR1 4 TERM
 0 0  1 0  0 1  1 1  0 2  1 2
 0 2.42  1 2.42  0 3.42  1 3.42  0 4.42  1 4.42
 1 2 4 3
 3 4 6 5
 7 8 10 9
 9 10 12 11
COMP COUL VERT LECT tous TERM
MATE LINE RO 8000.0 YOUN 1.E12 NU 0.0
          LECT tous TERM
INIT VITE 2  50 LECT 1 PAS 1 6 TERM
     VITE 2 -50 LECT 7 PAS 1 12 TERM
OPTI PINS DUMP
          STAT
         !VIDE
         !EQVL
         !EQVD
          ASN
         !NORB
LINK DECO PINB PENA SFAC 1.0
               BODY LECT 2 TERM
               BODY LECT 3 TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA CONT ECRO FREQ 1
     FICH ALIC FREQ 1
OPTI NOTE
 CSTA 0.5E0
 LOG 1
CALCUL TINI 0. TEND 100.E-3 NMAX 20
*==================================================================
PLAY
CAME   1 EYE   5.00000E-01  2.21000E+00  1.21286E+01
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
FREQ 2
GO
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB PARE
     COLO PAPE
```

```
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB NASN PASN
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB NORM
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
FREQ 20
GO
ENDPLAY
*==================================================================
SUIT
Post-treatment (time curves from alice file)
ECHO
RESU ALIC GARD PSCR
SORT GRAP
AXTE 1.0 'Time [s]'
COUR  1 'dy_5'  DEPL COMP 2 NOEU LECT  5 TERM
COUR  2 'dy_7'  DEPL COMP 2 NOEU LECT  7 TERM
COUR 11 'fy_5'  FEXT COMP 2 NOEU LECT  5 TERM
COUR 12 'fy_7'  FEXT COMP 2 NOEU LECT  7 TERM
TRAC 1 2 AXES 1.0 'DISPL. [M]' YZER
TRAC 11 12 AXES 1.0 'FEXT [N]' YZER
LIST 1 2 AXES 1.0 'DISPL. [M]' YZER
LIST 11 12 AXES 1.0 'FEXT [N]' YZER
*==================================================================
FIN
```

pene03.epx

```
PENE03
ECHO
!CONV win
TRID LAGR
GEOM LIBR POIN 24 CUBE 4 TERM
 0 0 0  1 0 0  0 1 0  1 1 0  0 2 0  1 2 0
 0 2.75 0  1 2.75 0  0 3.75 0  1 3.75 0  0 4.75 0  1 4.75 0
 0 0 1  1 0 1  0 1 1  1 1 1  0 2 1  1 2 1
 0 2.75 1  1 2.75 1  0 3.75 1  1 3.75 1  0 4.75 1  1 4.75 1
 1 2 4 3 13 14 16 15
 3 4 6 5 15 16 18 17
 7 8 10 9 19 20 22 21
 9 10 12 11 21 22 24 23
COMP COUL VERT LECT tous TERM
MATE LINE RO 8000.0 YOUN 1.E12 NU 0.0
          LECT tous TERM
INIT VITE 2  50 LECT 1 PAS 1 6 13 PAS 1 18 TERM
     VITE 2 -50 LECT 7 PAS 1 12 19 PAS 1 24 TERM
OPTI PINS DUMP
          STAT
         !VIDE
         !EQVL
         !EQVD
          ASN
         !NORB
LINK DECO PINB PENA SFAC 1.0
               BODY LECT 2 TERM
               BODY LECT 3 TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA CONT ECRO FREQ 1
     FICH ALIC FREQ 1
OPTI NOTE
 CSTA 0.5E0
 LOG 1
 dump
CALCUL TINI 0. TEND 100.E-3 NMAX 25
*==================================================================
PLAY
CAME   1 EYE   5.00000E-01  2.37500E+00  1.38610E+01
!        Q      1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
FREQ 5
GO
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB PARE
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB NASN PASN
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB NORM
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
FREQ 25
GO
ENDPLAY
*==================================================================
SUIT
Post-treatment (time curves from alice file)
ECHO
RESU ALIC GARD PSCR
SORT GRAP
AXTE 1.0 'Time [s]'
COUR  1 'dy_5'  DEPL COMP 2 NOEU LECT  5 TERM
COUR  2 'dy_7'  DEPL COMP 2 NOEU LECT  7 TERM
COUR 11 'fy_5'  FEXT COMP 2 NOEU LECT  5 TERM
COUR 12 'fy_7'  FEXT COMP 2 NOEU LECT  7 TERM
TRAC 1 2 AXES 1.0 'DISPL. [M]' YZER
TRAC 11 12 AXES 1.0 'FEXT [N]' YZER
LIST 1 2 AXES 1.0 'DISPL. [M]' YZER
LIST 11 12 AXES 1.0 'FEXT [N]' YZER
*==================================================================
FIN
```

pene04.epx

```
PENE04
ECHO
!CONV win
TRID LAGR
GEOM LIBR POIN 24 CUBE 4 TERM
 0 0 0  1 0 0  0 1 0  1 1 0  0 2 0  1 2 0
 0 2.75 0  1 2.75 0  0 3.75 0  1 3.75 0  0 4.75 0  1 4.75 0
 0 0 1  1 0 1  0 1 1  1 1 1  0 2 1  1 2 1
 0 2.75 1  1 2.75 1  0 3.75 1  1 3.75 1  0 4.75 1  1 4.75 1
 1 2 4 3 13 14 16 15
 3 4 6 5 15 16 18 17
 7 8 10 9 19 20 22 21
 9 10 12 11 21 22 24 23
COMP COUL VERT LECT tous TERM
MATE LINE RO 8000.0 YOUN 1.E12 NU 0.0
          LECT tous TERM
INIT VITE 2  50 LECT 1 PAS 1 6 13 PAS 1 18 TERM
     VITE 2 -50 LECT 7 PAS 1 12 19 PAS 1 24 TERM
OPTI PINS DUMP
          STAT
         !VIDE
         !EQVL
         !EQVD
         !NORB
LINK DECO PINB PENA SFAC 1.0
               BODY LECT 2 TERM
               BODY LECT 3 TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA CONT ECRO FREQ 1
     FICH ALIC FREQ 1
OPTI NOTE
 CSTA 0.5E0
 LOG 1
 dump
CALCUL TINI 0. TEND 100.E-3 NMAX 25
*==================================================================
PLAY
CAME   1 EYE   5.00000E-01  2.37500E+00  1.38610E+01
!        Q      1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
FREQ 5
GO
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB PARE
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB NASN PASN
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB NORM
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
FREQ 25
GO
ENDPLAY
*==================================================================
SUIT
Post-treatment (time curves from alice file)
ECHO
RESU ALIC GARD PSCR
SORT GRAP
AXTE 1.0 'Time [s]'
COUR  1 'dy_5'  DEPL COMP 2 NOEU LECT  5 TERM
COUR  2 'dy_7'  DEPL COMP 2 NOEU LECT  7 TERM
COUR 11 'fy_5'  FEXT COMP 2 NOEU LECT  5 TERM
COUR 12 'fy_7'  FEXT COMP 2 NOEU LECT  7 TERM
TRAC 1 2 AXES 1.0 'DISPL. [M]' YZER
TRAC 11 12 AXES 1.0 'FEXT [N]' YZER
LIST 1 2 AXES 1.0 'DISPL. [M]' YZER
LIST 11 12 AXES 1.0 'FEXT [N]' YZER
*==================================================================
FIN
```

pene05.eco

```
STANDARD VERSION

 TITLE: PENE05

      3>!CONV win
      4>TRID LAGR
      5>GEOM LIBR POIN 24 CUBE 4 TERM
 == SET FLAG IN INPUT FILE
 == GO BACK TO FLAG IN INPUT FILE
 == SET FLAG IN INPUT FILE
 == GO BACK TO FLAG IN INPUT FILE
      6> 0 0 0  1 0 0  0 1 0  1 1 0  0 2 0  1 2 0
      7> 0 2.75 0  1 2.75 0  0 3.75 0  1 3.75 0  0 4.75 0  1 4.75 0
      8> 0 0 1  1 0 1  0 1 1  1 1 1  0 2 1  1 2 1
      9> 0 2.75 1  1 2.75 1  0 3.75 1  1 3.75 1  0 4.75 1  1 4.75 1
     10> 1 2 4 3 13 14 16 15
     11> 3 4 6 5 15 16 18 17
     12> 7 8 10 9 19 20 22 21
     13> 9 10 12 11 21 22 24 23
     14>COMP COUL VERT LECT tous TERM
     15>MATE LINE RO 8000.0 YOUN 1.E12 NU 0.0
```

```
     16>          LECT tous TERM
     17>INIT VITE 2  50 LECT 1 PAS 1 6 13 PAS 1 18 TERM
     18>     VITE 2 -50 LECT 7 PAS 1 12 19 PAS 1 24 TERM
     19>OPTI PINS DUMP
     20>          STAT
     21>          VIDE
     22>
     22>!EQVL
     23>
     23>!EQVD
     24>          ASN
     25>
     25>!NORB
     26>LINK DECO PINB PENA SFAC 1.0
     27>               BODY MLEV 2 LECT 2 TERM
     28>               BODY MLEV 2 LECT 3 TERM
     29>ECRI COOR DEPL VITE ACCE FINT FEXT FLIA CONT ECRO FREQ 1
     30>     FICH ALIC FREQ 1
     31>OPTI NOTE
     32> CSTA 0.5E0
     33> LOG 1
     34> dump
     35>CALCUL TINI 0. TEND 100.E-3 NMAX 30
     36>*=============================================================
     37>PLAY

 -> END OF INITIALISATIONS ** TCPU =     0.02 SEC.

     38>CAME   1 EYE   8.08516E+00  6.89220E+00  1.05288E+01
     39>!        Q     9.23066E-01 -1.28822E-01  3.28793E-01  1.52476E-01
     40>         VIEW -5.67710E-01 -3.38089E-01 -7.50601E-01
     41>         RIGH  7.37293E-01  1.96779E-01 -6.46279E-01
     42>         UP   -3.66202E-01  9.20312E-01 -1.37557E-01
     43>         FOV   2.48819E+01
     44>CAME   2 EYE   3.53406E+00  4.18188E+00  4.51150E+00
     45>!        Q     9.23066E-01 -1.28822E-01  3.28793E-01  1.52476E-01
     46>         VIEW -5.67710E-01 -3.38089E-01 -7.50601E-01
     47>         RIGH  7.37293E-01  1.96779E-01 -6.46279E-01
     48>         UP   -3.66202E-01  9.20312E-01 -1.37557E-01
     49>         FOV   2.48819E+01
     50>SCEN GEOM NAVI FREE
     51>          FACE HFRO
     52>          PINB PARE NASN PASN
     53>     COLO PAPE
     54>SLER CAM1 1 NFRA 1
     55>TRAC OFFS FICH BMP REND
     56>SCEN GEOM NAVI FREE
     57>          FACE HFRO
     58>          PINB CDES DASN
     59>     COLO PAPE
     60>SLER CAM1 1 NFRA 1
     61>TRAC OFFS FICH BMP REND
     62>SCEN GEOM NAVI FREE
     63>          FACE HFRO
     64>          PINB DASN
     65>     COLO PAPE
     66>SLER CAM1 2 NFRA 1
     67>TRAC OFFS FICH BMP REND
     68>ENDPLAY
  >  END OF TIME STEP NO :       0 * T = 0.00000E+00 * TCPU =      0.06 SEC.

** ATTENTION       1 IN THE ROUTINE COMPUTE_PINBALL_CONTACTS ** THE MESSAGE IS T
HE FOLLOWING ONE :
CALCULATION STOPS HERE BECAUSE OF OPTI PINS VIDE

Saving image 0001 to pene05_0001.bmp ... Checking OpenGL errors ...
bitmap saved.
Saving image 0002 to pene05_0002.bmp ... Checking OpenGL errors ...
bitmap saved.
Saving image 0003 to pene05_0003.bmp ... Checking OpenGL errors ...
bitmap saved.
```

pene05.epx

```
PENE05
ECHO
!CONV win
TRID LAGR
GEOM LIBR POIN 24 CUBE 4 TERM
 0 0 0  1 0 0  0 1 0  1 1 0  0 2 0  1 2 0
 0 2.75 0  1 2.75 0  0 3.75 0  1 3.75 0  0 4.75 0  1 4.75 0
 0 0 1  1 0 1  0 1 1  1 1 1  0 2 1  1 2 1
 0 2.75 1  1 2.75 1  0 3.75 1  1 3.75 1  0 4.75 1  1 4.75 1
 1 2 4 3 13 14 16 15
 3 4 6 5 15 16 18 17
 7 8 10 9 19 20 22 21
 9 10 12 11 21 22 24 23
COMP COUL VERT LECT tous TERM
MATE LINE RO 8000.0 YOUN 1.E12 NU 0.0
          LECT tous TERM
INIT VITE 2  50 LECT 1 PAS 1 6 13 PAS 1 18 TERM
     VITE 2 -50 LECT 7 PAS 1 12 19 PAS 1 24 TERM
OPTI PINS DUMP
          STAT
          VIDE
         !EQVL
         !EQVD
          ASN
         !NORB
LINK DECO PINB PENA SFAC 1.0
               BODY MLEV 2 LECT 2 TERM
               BODY MLEV 2 LECT 3 TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA CONT ECRO FREQ 1
     FICH ALIC FREQ 1
OPTI NOTE
 CSTA 0.5E0
 LOG 1
 dump
CALCUL TINI 0. TEND 100.E-3 NMAX 30
*=============================================================
PLAY
CAME   1 EYE   8.08516E+00  6.89220E+00  1.05288E+01
!        Q     9.23066E-01 -1.28822E-01  3.28793E-01  1.52476E-01
         VIEW -5.67710E-01 -3.38089E-01 -7.50601E-01
         RIGH  7.37293E-01  1.96779E-01 -6.46279E-01
         UP   -3.66202E-01  9.20312E-01 -1.37557E-01
         FOV   2.48819E+01
CAME   2 EYE   3.53406E+00  4.18188E+00  4.51150E+00
!        Q     9.23066E-01 -1.28822E-01  3.28793E-01  1.52476E-01
         VIEW -5.67710E-01 -3.38089E-01 -7.50601E-01
         RIGH  7.37293E-01  1.96779E-01 -6.46279E-01
         UP   -3.66202E-01  9.20312E-01 -1.37557E-01
         FOV   2.48819E+01
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB PARE NASN PASN
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB CDES DASN
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB DASN
     COLO PAPE
SLER CAM1 2 NFRA 1
TRAC OFFS FICH BMP REND
ENDPLAY
*=============================================================
FIN
```

pene06.eco

```
STANDARD VERSION

 TITLE: PENE06

      3>!CONV win
      4>TRID LAGR
      5>GEOM LIBR POIN 24 CUBE 4 TERM
 == SET FLAG IN INPUT FILE
 == GO BACK TO FLAG IN INPUT FILE
 == SET FLAG IN INPUT FILE
 == GO BACK TO FLAG IN INPUT FILE
      6> 0 0 0  1 0 0  0 1 0  1 1 0  0 2 0  1 2 0
      7> 0 2.20 0  1 2.20 0  0 3.20 0  1 3.20 0  0 4.20 0  1 4.20 0
      8> 0 0 1  1 0 1  0 1 1  1 1 1  0 2 1  1 2 1
      9> 0 2.20 1  1 2.20 1  0 3.20 1  1 3.20 1  0 4.20 1  1 4.20 1
     10> 1 2 4 3 13 14 16 15
     11> 3 4 6 5 15 16 18 17
     12> 7 8 10 9 19 20 22 21
     13> 9 10 12 11 21 22 24 23
     14>COMP COUL VERT LECT tous TERM
     15>MATE LINE RO 8000.0 YOUN 1.E12 NU 0.0
     16>          LECT tous TERM
     17>INIT VITE 2  50 LECT 1 PAS 1 6 13 PAS 1 18 TERM
     18>     VITE 2 -50 LECT 7 PAS 1 12 19 PAS 1 24 TERM
     19>OPTI PINS DUMP
     20>          STAT
     21>
     21>!VIDE
     22>
     22>!EQVL
     23>
     23>!EQVD
     24>          ASN
     25>
     25>!NORB
     26>LINK DECO PINB PENA SFAC 1.0
     27>               BODY MLEV 2 LECT 2 TERM
     28>               BODY MLEV 2 LECT 3 TERM
     29>ECRI COOR DEPL VITE ACCE FINT FEXT FLIA CONT ECRO FREQ 1
     30>     FICH ALIC FREQ 1
     31>OPTI NOTE
     32> CSTA 0.5E0
     33> LOG 1
     34> dump
     35>CALCUL TINI 0. TEND 100.E-3 NMAX 30
     36>*=============================================================
     37>SUIT

 -> END OF INITIALISATIONS ** TCPU =     0.02 SEC.

  >  END OF TIME STEP NO :       0 * T = 0.00000E+00 * TCPU =      0.05 SEC.
  >  END OF TIME STEP NO :       1 * T = 3.87298E-05 * TCPU =      0.05 SEC.
  >  END OF TIME STEP NO :       2 * T = 7.74597E-05 * TCPU =      0.05 SEC.
  >  END OF TIME STEP NO :       3 * T = 1.16190E-04 * TCPU =      0.06 SEC.
  >  END OF TIME STEP NO :       4 * T = 1.54919E-04 * TCPU =      0.06 SEC.
  >  END OF TIME STEP NO :       5 * T = 1.93649E-04 * TCPU =      0.06 SEC.
  >  END OF TIME STEP NO :       6 * T = 2.32379E-04 * TCPU =      0.06 SEC.
  >  END OF TIME STEP NO :       7 * T = 2.71096E-04 * TCPU =      0.08 SEC.
  >  END OF TIME STEP NO :       8 * T = 3.09784E-04 * TCPU =      0.08 SEC.
  >  END OF TIME STEP NO :       9 * T = 3.48440E-04 * TCPU =      0.09 SEC.
  >  END OF TIME STEP NO :      10 * T = 3.87080E-04 * TCPU =      0.09 SEC.
  >  END OF TIME STEP NO :      11 * T = 4.25726E-04 * TCPU =      0.09 SEC.
  >  END OF TIME STEP NO :      12 * T = 4.64392E-04 * TCPU =      0.11 SEC.
  >  END OF TIME STEP NO :      13 * T = 5.03077E-04 * TCPU =      0.11 SEC.
  >  END OF TIME STEP NO :      14 * T = 5.41771E-04 * TCPU =      0.11 SEC.
  >  END OF TIME STEP NO :      15 * T = 5.80464E-04 * TCPU =      0.12 SEC.
  >  END OF TIME STEP NO :      16 * T = 6.19158E-04 * TCPU =      0.12 SEC.
  >  END OF TIME STEP NO :      17 * T = 6.57865E-04 * TCPU =      0.14 SEC.
  >  END OF TIME STEP NO :      18 * T = 6.96584E-04 * TCPU =      0.14 SEC.
  >  END OF TIME STEP NO :      19 * T = 7.35292E-04 * TCPU =      0.14 SEC.
  >  END OF TIME STEP NO :      20 * T = 7.73999E-04 * TCPU =      0.14 SEC.
  >  END OF TIME STEP NO :      21 * T = 8.12707E-04 * TCPU =      0.14 SEC.
  >  END OF TIME STEP NO :      22 * T = 8.51415E-04 * TCPU =      0.14 SEC.
  >  END OF TIME STEP NO :      23 * T = 8.90118E-04 * TCPU =      0.14 SEC.
  >  END OF TIME STEP NO :      24 * T = 9.28822E-04 * TCPU =      0.14 SEC.
  >  END OF TIME STEP NO :      25 * T = 9.67531E-04 * TCPU =      0.16 SEC.
  >  END OF TIME STEP NO :      26 * T = 1.00623E-03 * TCPU =      0.16 SEC.
  >  END OF TIME STEP NO :      27 * T = 1.04492E-03 * TCPU =      0.16 SEC.
  >  END OF TIME STEP NO :      28 * T = 1.08363E-03 * TCPU =      0.16 SEC.
  >  END OF TIME STEP NO :      29 * T = 1.12235E-03 * TCPU =      0.16 SEC.
  >  END OF TIME STEP NO :      30 * T = 1.16106E-03 * TCPU =      0.16 SEC.
     38>Post-treatment (time curves from alice file)

 TITLE: POST-TREATMENT (TIME CURVES FROM ALICE FILE)
```

```
     39>ECHO
     40>RESU ALIC GARD PSCR
     41>SORT GRAP
EDITION OF "ALICE" FILE fort.10 (VERSION 15) WITH THE TITLE:
 PENE06
THE FILE CREATED ON: 06-06-2014
 ALICE FILE N 10 STORAGE N    1 AT STEP       0 TIME  0.00000E+00
 ALICE FILE N 10 STORAGE N    2 AT STEP       1 TIME  3.87298E-05
 ALICE FILE N 10 STORAGE N    3 AT STEP       2 TIME  7.74597E-05
 ALICE FILE N 10 STORAGE N    4 AT STEP       3 TIME  1.16189E-04
 ALICE FILE N 10 STORAGE N    5 AT STEP       4 TIME  1.54919E-04
 ALICE FILE N 10 STORAGE N    6 AT STEP       5 TIME  1.93649E-04
 ALICE FILE N 10 STORAGE N    7 AT STEP       6 TIME  2.32379E-04
 ALICE FILE N 10 STORAGE N    8 AT STEP       7 TIME  2.71096E-04
 ALICE FILE N 10 STORAGE N    9 AT STEP       8 TIME  3.09784E-04
 ALICE FILE N 10 STORAGE N   10 AT STEP       9 TIME  3.48440E-04
 ALICE FILE N 10 STORAGE N   11 AT STEP      10 TIME  3.87080E-04
 ALICE FILE N 10 STORAGE N   12 AT STEP      11 TIME  4.25726E-04
 ALICE FILE N 10 STORAGE N   13 AT STEP      12 TIME  4.64392E-04
 ALICE FILE N 10 STORAGE N   14 AT STEP      13 TIME  5.03077E-04
 ALICE FILE N 10 STORAGE N   15 AT STEP      14 TIME  5.41771E-04
 ALICE FILE N 10 STORAGE N   16 AT STEP      15 TIME  5.80464E-04
 ALICE FILE N 10 STORAGE N   17 AT STEP      16 TIME  6.19158E-04
 ALICE FILE N 10 STORAGE N   18 AT STEP      17 TIME  6.57865E-04
 ALICE FILE N 10 STORAGE N   19 AT STEP      18 TIME  6.96584E-04
 ALICE FILE N 10 STORAGE N   20 AT STEP      19 TIME  7.35292E-04
 ALICE FILE N 10 STORAGE N   21 AT STEP      20 TIME  7.73999E-04
 ALICE FILE N 10 STORAGE N   22 AT STEP      21 TIME  8.12707E-04
 ALICE FILE N 10 STORAGE N   23 AT STEP      22 TIME  8.51415E-04
 ALICE FILE N 10 STORAGE N   24 AT STEP      23 TIME  8.90118E-04
 ALICE FILE N 10 STORAGE N   25 AT STEP      24 TIME  9.28822E-04
 ALICE FILE N 10 STORAGE N   26 AT STEP      25 TIME  9.67531E-04
 ALICE FILE N 10 STORAGE N   27 AT STEP      26 TIME  1.00623E-03
 ALICE FILE N 10 STORAGE N   28 AT STEP      27 TIME  1.04492E-03
 ALICE FILE N 10 STORAGE N   29 AT STEP      28 TIME  1.08363E-03
 ALICE FILE N 10 STORAGE N   30 AT STEP      29 TIME  1.12235E-03
 ALICE FILE N 10 STORAGE N   31 AT STEP      30 TIME  1.16106E-03
FILE CONTAINS :     31 TIME INSTANTS.
TMIN :    0.00000E+00  TMAX :     1.16106E-03
 == SET FLAG IN INPUT FILE
 == GO BACK TO FLAG IN INPUT FILE
     42>AXTE 1.0 'Time [s]'
     43>COUR  1 'dy_5'  DEPL COMP 2 NOEU LECT  5 TERM
     44>COUR  2 'dy_7'  DEPL COMP 2 NOEU LECT  7 TERM
     45>COUR 11 'fy_5'  FEXT COMP 2 NOEU LECT  5 TERM
     46>COUR 12 'fy_7'  FEXT COMP 2 NOEU LECT  7 TERM
     47>TRAC 1 2 AXES 1.0 'DISPL. [M]' YZER
     48>TRAC 11 12 AXES 1.0 'FEXT [N]' YZER
     49>LIST 1 2 AXES 1.0 'DISPL. [M]' YZER
     50>LIST 11 12 AXES 1.0 'FEXT [N]' YZER
     51>*=============================================================
     52>FIN
   >  TOTAL CPU TIME :        0.16 SEC.
```

pene06.epx

```
PENE06
ECHO
!CONV win
TRID LAGR
GEOM LIBR POIN 24 CUBE 4 TERM
 0 0 0  1 0 0  0 1 0  1 1 0  0 2 0  1 2 0
 0 2.20 0  1 2.20 0  0 3.20 0  1 3.20 0  0 4.20 0  1 4.20 0
 0 0 1  1 0 1  0 1 1  1 1 1  0 2 1  1 2 1
 0 2.20 1  1 2.20 1  0 3.20 1  1 3.20 1  0 4.20 1  1 4.20 1
 1 2 4 3 13 14 16 15
 3 4 6 5 15 16 18 17
 7 8 10 9 19 20 22 21
 9 10 12 11 21 22 24 23
COMP COUL VERT LECT tous TERM
MATE LINE RO 8000.0 YOUN 1.E12 NU 0.0
          LECT tous TERM
INIT VITE 2  50 LECT 1 PAS 1 6 13 PAS 1 18 TERM
     VITE 2 -50 LECT 7 PAS 1 12 19 PAS 1 24 TERM
OPTI PINS DUMP
          STAT
         !VIDE
         !EQVL
         !EQVD
          ASN
         !NORB
LINK DECO PINB PENA SFAC 1.0
               BODY MLEV 2 LECT 2 TERM
               BODY MLEV 2 LECT 3 TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA CONT ECRO FREQ 1
     FICH ALIC FREQ 1
OPTI NOTE
 CSTA 0.5E0
 LOG 1
 dump
CALCUL TINI 0. TEND 100.E-3 NMAX 30
*=============================================================
SUIT
Post-treatment (time curves from alice file)
ECHO
RESU ALIC GARD PSCR
SORT GRAP
AXTE 1.0 'Time [s]'
COUR  1 'dy_5'  DEPL COMP 2 NOEU LECT  5 TERM
COUR  2 'dy_7'  DEPL COMP 2 NOEU LECT  7 TERM
COUR 11 'fy_5'  FEXT COMP 2 NOEU LECT  5 TERM
COUR 12 'fy_7'  FEXT COMP 2 NOEU LECT  7 TERM
TRAC 1 2 AXES 1.0 'DISPL. [M]' YZER
TRAC 11 12 AXES 1.0 'FEXT [N]' YZER
LIST 1 2 AXES 1.0 'DISPL. [M]' YZER
LIST 11 12 AXES 1.0 'FEXT [N]' YZER
*=============================================================
FIN
```

pene07.dgibi

```
opti echo 1;
opti dime 2 elem qua4;
opti sauv form 'pene07.msh';
opti trac psc ftra 'pene07_mesh.ps';
p1 = 0 0;
p2 = 1 0;
p3 = 1 1;
p4 = 0 1;
e1 = manu qua4 p1 p2 p3 p4;
e2 = e1 plus (0 1.25);
depl e2 tour 45 (bary e2);
mesh = e1 et e2;
tass mesh;
sauv form mesh;
trac qual mesh;
fin;
```

pene07.epx

```
PENE07
ECHO
!CONV win
CAST mesh
CPLA LAGR
GEOM CAR1 mesh TERM
COMP COUL VERT LECT tous TERM
MATE LINE RO 8000.0 YOUN 1.E12 NU 0.0
          LECT tous TERM
INIT VITE 2  50 LECT e1 TERM
     VITE 2 -50 LECT e2 TERM
OPTI PINS DUMP
          STAT
         !VIDE
         !EQVL
         !EQVD
          ASN
          NORB
LINK DECO PINB PENA SFAC 1.0
               BODY MLEV 2 LECT e1 TERM
               BODY MLEV 2 LECT e2 TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA CONT ECRO FREQ 1
     FICH ALIC FREQ 1
OPTI NOTE
 CSTA 0.5E0
 LOG 1
CALCUL TINI 0. TEND 100.E-3 NMAX 30
*=============================================================
PLAY
CAME   1 EYE   5.00000E-01  2.21000E+00  1.21286E+01
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
FREQ 2
GO
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB PARE
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB NASN PASN
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB NORM
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
FREQ 30
GO
ENDPLAY
*=============================================================
SUIT
Post-treatment (time curves from alice file)
ECHO
RESU ALIC GARD PSCR
SORT GRAP
AXTE 1.0 'Time [s]'
COUR  1 'dy_4'  DEPL COMP 2 NOEU LECT  4 TERM
COUR  2 'dy_5'  DEPL COMP 2 NOEU LECT  5 TERM
COUR 11 'fy_4'  FEXT COMP 2 NOEU LECT  4 TERM
COUR 12 'fy_5'  FEXT COMP 2 NOEU LECT  5 TERM
TRAC 1 2 AXES 1.0 'DISPL. [M]' YZER
TRAC 11 12 AXES 1.0 'FEXT [N]' YZER
LIST 1 2 AXES 1.0 'DISPL. [M]' YZER
LIST 11 12 AXES 1.0 'FEXT [N]' YZER
*=============================================================
FIN
```

pene08.dgibi

```
opti echo 1;
opti dime 3 elem cub8;
opti sauv form 'pene08.msh';
opti trac psc ftra 'pene08_mesh.ps';
p1 = 0 0 0;
p2 = 1 0 0;
p3 = 1 1 0;
p4 = 0 1 0;
p5 = 0 0 1;
p6 = 1 0 1;
p7 = 1 1 1;
p8 = 0 1 1;
e1 = manu cub8 p1 p2 p3 p4 p5 p6 p7 p8;
e2 = e1 plus (0 0 1.425);
be2 = bary e2;
be2p = be2 plus (0 0 1);
depl e2 tour 45 be2 be2p;
be2q = be2 plus (1 0 0);
```

```
depl e2 tour 54.7356 be2 be2q;
mesh = e1 et e2;
tass mesh;
sauv form mesh;
oeil = 100000 200000 300000;
trac oeil cach qual mesh;
fin;
```

pene08.epx

```
PENE08
ECHO
!CONV win
CAST mesh
TRID LAGR
GEOM CUBE mesh TERM
COMP COUL VERT LECT tous TERM
MATE LINE RO 8000.0 YOUN 1.E12 NU 0.0
          LECT tous TERM
INIT VITE 3  50 LECT e1 TERM
     VITE 3 -50 LECT e2 TERM
OPTI PINS DUMP
          STAT
         !VIDE
         !EQVL
         !EQVD
          ASN
          NORB
LINK DECO PINB PENA SFAC 1.0
               BODY MLEV 2 LECT e1 TERM
               BODY MLEV 2 LECT e2 TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA CONT ECRO FREQ 1
     FICH ALIC FREQ 1
OPTI NOTE
 CSTA 0.5E0
 LOG 1
CALCUL TINI 0. TEND 100.E-3 NMAX 30
*==================================================================
PLAY
CAME   1 EYE   5.61066E+00 -7.84855E+00  3.20660E+00
!        Q     7.59557E-01  5.93659E-01  2.18543E-01  1.51267E-01
         VIEW -5.11594E-01  8.35718E-01 -1.99616E-01
         RIGH  8.58714E-01  4.89272E-01 -1.52390E-01
         UP    2.96888E-02  2.49375E-01  9.67952E-01
         FOV   2.48819E+01
FREQ 2
GO
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB PARE
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB NASN PASN
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB NORM
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
FREQ 30
GO
ENDPLAY
*==================================================================
SUIT
Post-treatment (time curves from alice file)
ECHO
RESU ALIC GARD PSCR
SORT GRAP
AXTE 1.0 'Time [s]'
COUR  1 'dz_5'  DEPL COMP 3 NOEU LECT  5 TERM
COUR  2 'dz_9'  DEPL COMP 3 NOEU LECT  9 TERM
COUR 11 'fz_5'  FEXT COMP 3 NOEU LECT  5 TERM
COUR 12 'fz_9'  FEXT COMP 3 NOEU LECT  9 TERM
TRAC 1 2 AXES 1.0 'DISPL. [M]' YZER
TRAC 11 12 AXES 1.0 'FEXT [N]' YZER
LIST 1 2 AXES 1.0 'DISPL. [M]' YZER
LIST 11 12 AXES 1.0 'FEXT [N]' YZER
*==================================================================
FIN
```

pene09.dgibi

```
opti echo 1;
opti dime 3 elem cub8;
opti sauv form 'pene09.msh';
opti trac psc ftra 'pene09_mesh.ps';
p1 = 0 0 0;
p2 = 1 0 0;
p3 = 1 1 0;
p4 = 0 1 0;
p5 = 0 0 1;
p6 = 1 0 1;
p7 = 1 1 1;
p8 = 0 1 1;
e1 = manu cub8 p1 p2 p3 p4 p5 p6 p7 p8;
e2 = e1 plus (0 0 1.325);
be2 = bary e2;
be2p = be2 plus (0 (0-1) 0);
depl e2 tour 45 be2 be2p;
mesh = e1 et e2;
tass mesh;
sauv form mesh;
oeil = 100000 200000 300000;
trac oeil cach qual mesh;
fin;
```

pene09.epx

```
PENE09
ECHO
!CONV win
CAST mesh
TRID LAGR
GEOM CUBE mesh TERM
COMP COUL VERT LECT tous TERM
MATE LINE RO 8000.0 YOUN 1.E12 NU 0.0
          LECT tous TERM
INIT VITE 3  50 LECT e1 TERM
     VITE 3 -50 LECT e2 TERM
OPTI PINS DUMP
          STAT
         !VIDE
         !EQVL
         !EQVD
          ASN
          NORB
LINK DECO PINB PENA SFAC 1.0
               BODY MLEV 2 LECT e1 TERM
               BODY MLEV 2 LECT e2 TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA CONT ECRO FREQ 1
     FICH ALIC FREQ 1
OPTI NOTE
 CSTA 0.5E0
 LOG 1
CALCUL TINI 0. TEND 100.E-3 NMAX 30
*==================================================================
PLAY
CAME   1 EYE   5.61066E+00 -7.84855E+00  3.20660E+00
!        Q     7.59557E-01  5.93659E-01  2.18543E-01  1.51267E-01
         VIEW -5.11594E-01  8.35718E-01 -1.99616E-01
         RIGH  8.58714E-01  4.89272E-01 -1.52390E-01
         UP    2.96888E-02  2.49375E-01  9.67952E-01
         FOV   2.48819E+01
FREQ 2
GO
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB PARE
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB NASN PASN
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB NORM
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
FREQ 30
GO
ENDPLAY
*==================================================================
SUIT
Post-treatment (time curves from alice file)
ECHO
RESU ALIC GARD PSCR
SORT GRAP
AXTE 1.0 'Time [s]'
COUR  1 'dz_5'  DEPL COMP 3 NOEU LECT  5 TERM
COUR  2 'dz_9'  DEPL COMP 3 NOEU LECT  9 TERM
COUR 11 'fz_5'  FEXT COMP 3 NOEU LECT  5 TERM
COUR 12 'fz_9'  FEXT COMP 3 NOEU LECT  9 TERM
TRAC 1 2 AXES 1.0 'DISPL. [M]' YZER
TRAC 11 12 AXES 1.0 'FEXT [N]' YZER
LIST 1 2 AXES 1.0 'DISPL. [M]' YZER
LIST 11 12 AXES 1.0 'FEXT [N]' YZER
*==================================================================
FIN
```

pene10.dgibi

```
opti echo 1;
opti dime 3 elem cub8;
opti sauv form 'pene10.msh';
opti trac psc ftra 'pene10_mesh.ps';
p1 = 0 0 0;
p2 = 1 0 0;
p3 = 1 1 0;
p4 = 0 1 0;
p5 = 0 0 1;
p6 = 1 0 1;
p7 = 1 1 1;
p8 = 0 1 1;
e1 = manu cub8 p1 p2 p3 p4 p5 p6 p7 p8;
e2 = e1 plus (0 0 1.460);
be2 = bary e2;
be2p = be2 plus (0 (0-1) 0);
depl e2 tour 45 be2 be2p;
be1 = bary e1;
be1p = be1 plus (1 0 0);
depl e1 tour 45 be1 be1p;
mesh = e1 et e2;
tass mesh;
sauv form mesh;
oeil = 100000 200000 300000;
trac oeil cach qual mesh;
fin;
```

pene10.epx

```
PENE10
ECHO
!CONV win
```

```
CAST mesh
TRID LAGR
GEOM CUBE mesh TERM
COMP COUL VERT LECT tous TERM
MATE LINE RO 8000.0 YOUN 1.E12 NU 0.0
          LECT tous TERM
INIT VITE 3  50 LECT e1 TERM
     VITE 3 -50 LECT e2 TERM
OPTI PINS DUMP
          STAT
         !VIDE
         !EQVL
         !EQVD
          ASN
          NORB
LINK DECO PINB PENA SFAC 1.0
               BODY MLEV 2 LECT e1 TERM
               BODY MLEV 2 LECT e2 TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA CONT ECRO FREQ 1
     FICH ALIC FREQ 1
OPTI NOTE
 CSTA 0.5E0
 LOG 1
CALCUL TINI 0. TEND 100.E-3 NMAX 30
*=================================================================
PLAY
CAME   1 EYE   5.61066E+00 -7.84855E+00  3.20660E+00
!        Q     7.59557E-01  5.93659E-01  2.18543E-01  1.51267E-01
         VIEW -5.11594E-01  8.35718E-01 -1.99616E-01
         RIGH  8.58714E-01  4.89272E-01 -1.52390E-01
         UP    2.96888E-02  2.49375E-01  9.67952E-01
         FOV   2.48819E+01
FREQ 2
GO
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB PARE
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB NASN PASN
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB NORM
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
FREQ 30
GO
ENDPLAY
*=================================================================
SUIT
Post-treatment (time curves from alice file)
ECHO
RESU ALIC GARD PSCR
SORT GRAP
AXTE 1.0 'Time [s]'
COUR  1 'dz_7'  DEPL COMP 3 NOEU LECT  7 TERM
COUR  2 'dz_9'  DEPL COMP 3 NOEU LECT  9 TERM
COUR 11 'fz_7'  FEXT COMP 3 NOEU LECT  7 TERM
COUR 12 'fz_9'  FEXT COMP 3 NOEU LECT  9 TERM
TRAC 1 2 AXES 1.0 'DISPL. [M]' YZER
TRAC 11 12 AXES 1.0 'FEXT [N]' YZER
LIST 1 2 AXES 1.0 'DISPL. [M]' YZER
LIST 11 12 AXES 1.0 'FEXT [N]' YZER
*=================================================================
FIN
```

pmat01.epx

```
PMAT01
ECHO
!CONV win
CPLA LAGR
GEOM LIBR POIN 2 PMAT 2 TERM
 -0.55 0  0.55 0
  1
  2
COMP EPAI 1.0 LECT tous TERM
     COUL VERT LECT tous TERM
MATE LINE RO 8000.0 YOUN 1.E12 NU 0.0
          LECT tous TERM
INIT VITE 1  50 LECT 1 TERM
     VITE 1 -50 LECT 2 TERM
OPTI PINS DUMP
          STAT
          ASN
LINK DECO PINB PENA SFAC 1.0
               BODY DIAM 1.0 LECT 1 TERM
               BODY DIAM 1.0 LECT 2 TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA FDEC CONT ECRO FREQ 50
     FICH ALIC FREQ 1
OPTI NOTE
 CSTA 0.5E0
 LOG 1
 DPMA
CALC TINI 0. TEND 100.E-3 NMAX 100
*=================================================================
PLAY
CAME   1 EYE   0.00000E+00  0.00000E+00  6.32949E+00
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  0.00000E+00  0.00000E+00  0.00000E+00
!RSPHERE:  1.26590E+00
!RADIUS :  6.32949E+00
!ASPECT :  1.00000E+00
!NEAR   :  4.93701E+00
!FAR    :  8.86129E+00
SCEN GEOM NAVI FREE
          PINB CDES
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH AVI NOCL NFTO 101 FPS 10 KFRE 10 COMP -1 REND
FREQ 1
GOTR LOOP 99 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDPLAY
*=================================================================
SUIT
Post treatment from ALIC file
ECHO
RESU ALIC GARD PSCR
SORT GRAP
AXTE 1.0 'Time [s]'
COUR 1 'dx_1' DEPL COMP 1 NOEU LECT 1 TERM
COUR 2 'dx_2' DEPL COMP 1 NOEU LECT 2 TERM
COUR 3 'vx_1' VITE COMP 1 NOEU LECT 1 TERM
COUR 4 'vx_2' VITE COMP 1 NOEU LECT 2 TERM
COUR 5 'fx_1' FEXT COMP 1 NOEU LECT 1 TERM
COUR 6 'fx_2' FEXT COMP 1 NOEU LECT 2 TERM
TRAC 1 2 AXES 1.0 'DISPL. [M]' YZER
TRAC 3 4 AXES 1.0 'VELOC. [M/S]' YZER
TRAC 5 6 AXES 1.0 'FEXT. [N]' YZER
*=================================================================
FIN
```

pmat02.dgibi

```
opti echo 1;
opti dime 3 elem cub8;
opti sauv form 'pmat02.msh';
opti trac psc ftra 'pmat02_mesh.ps';
p1 = -1 -1 0;
p2 =  1 -1 0;
p3 =  1  1 0;
p4 = -1  1 0;
n = 9;
c1 = p1 d n p2;
c2 = p2 d n p3;
c3 = p3 d n p4;
c4 = p4 d n p1;
plate = dall c1 c2 c3 c4 plan;
pm = 0 0 0.12;
pmt = manu poi1 pm;
mesh = plate et pmt;
tass mesh;
sauv form mesh;
trac cach qual mesh;
fin;
```

pmat02.epx

```
PMAT02
ECHO
!CONV win
CAST mesh
TRID LAGR
GEOM Q4GS plate PMAT pmt TERM
COMP EPAI 0.01 LECT plate TERM
          0.18 LECT pmt TERM
     COUL VERT LECT plate TERM
          ROUG LECT pmt TERM
MATE LINE RO 8000.0 YOUN 2.E11 NU 0.3
          LECT tous TERM
INIT VITE 3 -100 LECT pmt TERM
OPTI PINS DUMP
          STAT
          ASN
LINK DECO BLOQ 123456 LECT c1 c2 c3 c4 TERM
          PINB PENA SFAC 1.0
               BODY LECT plate TERM
               BODY DIAM 0.18 LECT pmt TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA FDEC CONT ECRO TFRE 1.E-3
     FICH ALIC FREQ 1
OPTI NOTE
 CSTA 0.5E0
 LOG 1
 DPMA
CALC TINI 0. TEND 5.E-3
*=================================================================
PLAY
CAME   1 EYE   0.00000E+00 -4.99864E+00  2.35734E+00
!        Q     8.43391E-01  5.37300E-01  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  9.06308E-01 -4.22618E-01
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  4.22618E-01  9.06308E-01
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  0.00000E+00  0.00000E+00  2.64326E-02
!RSPHERE:  1.49065E+00
!RADIUS :  5.51539E+00
!ASPECT :  1.00000E+00
!NEAR   :  4.02475E+00
!FAR    :  8.49669E+00
SCEN GEOM NAVI FREE
          PINB CDES
     COLO PAPE
     LIMA ON
SLER CAM1 1 NFRA 1
TRAC OFFS FICH AVI NOCL NFTO 557 FPS 25 KFRE 10 COMP -1 REND
FREQ 1
GOTR LOOP 555 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDPLAY
*=================================================================
SUIT
Post treatment from ALIC file
ECHO
```

```
RESU ALIC GARD PSCR
SORT GRAP
AXTE 1.0 'Time [s]'
COUR 1 'dz_pmt' DEPL COMP 3 NOEU LECT pmt TERM
COUR 2 'dx_pmt' DEPL COMP 1 NOEU LECT pmt TERM
COUR 3 'vz_pmt' VITE COMP 3 NOEU LECT pmt TERM
COUR 5 'fz_pmt' FEXT COMP 3 NOEU LECT pmt TERM
TRAC 1 AXES 1.0 'DISPL. [M]' YZER
TRAC 3 AXES 1.0 'VELOC. [M/S]' YZER
TRAC 5 AXES 1.0 'FEXT. [N]' YZER
TRAC 2 AXES 1.0 'DISPL. [M]' YZER
LIST 1 AXES 1.0 'DISPL. [M]' YZER
LIST 3 AXES 1.0 'VELOC. [M/S]' YZER
LIST 5 AXES 1.0 'FEXT. [N]' YZER
LIST 2 AXES 1.0 'DISPL. [M]' YZER
*=================================================================
FIN
```

pmat03.dgibi

```
opti echo 1;
opti dime 3 elem cub8;
opti sauv form 'pmat03.msh';
opti trac psc ftra 'pmat03_mesh.ps';
p1 = -1 -1 0;
p2 =  1 -1 0;
p3 =  1  1 0;
p4 = -1  1 0;
n = 10;
c1 = p1 d n p2;
c2 = p2 d n p3;
c3 = p3 d n p4;
c4 = p4 d n p1;
plate = dall c1 c2 c3 c4 plan;
pm = 0 0 0.12;
pmt = manu poi1 pm;
mesh = plate et pmt;
tass mesh;
sauv form mesh;
trac cach qual mesh;
fin;
```

pmat03.epx

```
PMAT03
ECHO
!CONV win
CAST mesh
TRID LAGR
GEOM Q4GS plate PMAT pmt TERM
COMP EPAI 0.01 LECT plate TERM
          0.18 LECT pmt TERM
     COUL VERT LECT plate TERM
          ROUG LECT pmt TERM
MATE LINE RO 8000.0 YOUN 2.E11 NU 0.3
          LECT tous TERM
INIT VITE 3 -100 LECT pmt TERM
OPTI PINS DUMP
          STAT
          ASN
LINK DECO BLOQ 123456 LECT c1 c2 c3 c4 TERM
          PINB PENA SFAC 1.0
               BODY LECT plate TERM
               BODY DIAM 0.18 LECT pmt TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA FDEC CONT ECRO TFRE 1.E-3
     FICH ALIC FREQ 1
OPTI NOTE
 CSTA 0.5E0
 LOG 1
 DPMA
CALC TINI 0. TEND 5.E-3
*=================================================================
PLAY
CAME   1 EYE   0.00000E+00 -4.99864E+00  2.35734E+00
!        Q     8.43391E-01  5.37300E-01  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  9.06308E-01 -4.22618E-01
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  4.22618E-01  9.06308E-01
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  0.00000E+00  0.00000E+00  2.64326E-02
!RSPHERE:  1.49065E+00
!RADIUS :  5.51539E+00
!ASPECT :  1.00000E+00
!NEAR   :  4.02475E+00
!FAR    :  8.49669E+00
SCEN GEOM NAVI FREE
          PINB CDES
     COLO PAPE
     LIMA ON
SLER CAM1 1 NFRA 1
TRAC OFFS FICH AVI NOCL NFTO 557 FPS 25 KFRE 10 COMP -1 REND
FREQ 1
GOTR LOOP 555 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDPLAY
*=================================================================
SUIT
Post treatment from ALIC file
ECHO
RESU ALIC GARD PSCR
SORT GRAP
AXTE 1.0 'Time [s]'
COUR 1 'dz_pmt' DEPL COMP 3 NOEU LECT pmt TERM
COUR 2 'dx_pmt' DEPL COMP 1 NOEU LECT pmt TERM
COUR 3 'vz_pmt' VITE COMP 3 NOEU LECT pmt TERM
COUR 5 'fz_pmt' FEXT COMP 3 NOEU LECT pmt TERM
RCOU 11 'dz_pmt' FICH 'pmat02.pun' RENA 'dz_pmt_02'
RCOU 13 'vz_pmt' FICH 'pmat02.pun' RENA 'vz_pmt_02'
RCOU 15 'fz_pmt' FICH 'pmat02.pun' RENA 'fz_pmt_02'
TRAC 1 AXES 1.0 'DISPL. [M]' YZER
TRAC 3 AXES 1.0 'VELOC. [M/S]' YZER
TRAC 5 AXES 1.0 'FEXT. [N]' YZER
TRAC 2 AXES 1.0 'DISPL. [M]' YZER
LIST 1 AXES 1.0 'DISPL. [M]' YZER
LIST 3 AXES 1.0 'VELOC. [M/S]' YZER
LIST 5 AXES 1.0 'FEXT. [N]' YZER
LIST 2 AXES 1.0 'DISPL. [M]' YZER
TRAC 1 11 AXES 1.0 'DISPL. [M]' YZER
COLO NOIR ROUG
TRAC 3 13 AXES 1.0 'VELOC. [M/S]' YZER
COLO NOIR ROUG
TRAC 5 15 AXES 1.0 'FEXT. [N]' YZER
COLO NOIR ROUG
*=================================================================
FIN
```

pmat04.dgibi

```
opti echo 1;
opti dime 3 elem cub8;
opti sauv form 'pmat04.msh';
opti trac psc ftra 'pmat04_mesh.ps';
p1 = -1 -1 0;
p2 =  1 -1 0;
p3 =  1  1 0;
p4 = -1  1 0;
n = 19;
c1 = p1 d n p2;
c2 = p2 d n p3;
c3 = p3 d n p4;
c4 = p4 d n p1;
plate = dall c1 c2 c3 c4 plan;
pm = 0 0 0.12;
pmt = manu poi1 pm;
mesh = plate et pmt;
tass mesh;
sauv form mesh;
trac cach qual mesh;
fin;
```

pmat04.epx

```
PMAT04
ECHO
!CONV win
CAST mesh
TRID LAGR
GEOM Q4GS plate PMAT pmt TERM
COMP EPAI 0.01 LECT plate TERM
          0.18 LECT pmt TERM
     COUL VERT LECT plate TERM
          ROUG LECT pmt TERM
MATE LINE RO 8000.0 YOUN 2.E11 NU 0.3
          LECT tous TERM
INIT VITE 3 -100 LECT pmt TERM
OPTI PINS DUMP
          STAT
          ASN
LINK DECO BLOQ 123456 LECT c1 c2 c3 c4 TERM
          PINB PENA SFAC 1.0
               BODY LECT plate TERM
               BODY DIAM 0.18 LECT pmt TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA FDEC CONT ECRO TFRE 1.E-3
     FICH ALIC FREQ 1
OPTI NOTE
 CSTA 0.5E0
 LOG 1
 DPMA
CALC TINI 0. TEND 5.E-3
*=================================================================
PLAY
CAME   1 EYE   0.00000E+00 -4.99864E+00  2.35734E+00
!        Q     8.43391E-01  5.37300E-01  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  9.06308E-01 -4.22618E-01
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  4.22618E-01  9.06308E-01
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  0.00000E+00  0.00000E+00  2.64326E-02
!RSPHERE:  1.49065E+00
!RADIUS :  5.51539E+00
!ASPECT :  1.00000E+00
!NEAR   :  4.02475E+00
!FAR    :  8.49669E+00
SCEN GEOM NAVI FREE
          PINB CDES
     COLO PAPE
     LIMA ON
SLER CAM1 1 NFRA 1
TRAC OFFS FICH AVI NOCL NFTO 557 FPS 25 KFRE 10 COMP -1 REND
FREQ 1
GOTR LOOP 555 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDPLAY
*=================================================================
SUIT
Post treatment from ALIC file
ECHO
RESU ALIC GARD PSCR
SORT GRAP
AXTE 1.0 'Time [s]'
COUR 1 'dz_pmt' DEPL COMP 3 NOEU LECT pmt TERM
COUR 2 'dx_pmt' DEPL COMP 1 NOEU LECT pmt TERM
COUR 3 'vz_pmt' VITE COMP 3 NOEU LECT pmt TERM
COUR 5 'fz_pmt' FEXT COMP 3 NOEU LECT pmt TERM
TRAC 1 AXES 1.0 'DISPL. [M]' YZER
TRAC 3 AXES 1.0 'VELOC. [M/S]' YZER
TRAC 5 AXES 1.0 'FEXT. [N]' YZER
TRAC 2 AXES 1.0 'DISPL. [M]' YZER
LIST 1 AXES 1.0 'DISPL. [M]' YZER
LIST 3 AXES 1.0 'VELOC. [M/S]' YZER
LIST 5 AXES 1.0 'FEXT. [N]' YZER
LIST 2 AXES 1.0 'DISPL. [M]' YZER
```

```
*================================================================
FIN
```

pmat05.dgibi

```
opti echo 1;
opti dime 3 elem cub8;
opti sauv form 'pmat05.msh';
opti trac psc ftra 'pmat05_mesh.ps';
p1 = -1 -1 0;
p2 =  1 -1 0;
p3 =  1  1 0;
p4 = -1  1 0;
n = 20;
c1 = p1 d n p2;
c2 = p2 d n p3;
c3 = p3 d n p4;
c4 = p4 d n p1;
plate = dall c1 c2 c3 c4 plan;
pm = 0 0 0.12;
pmt = manu poi1 pm;
mesh = plate et pmt;
tass mesh;
sauv form mesh;
trac cach qual mesh;
fin;
```

pmat05.epx

```
PMAT05
ECHO
!CONV win
CAST mesh
TRID LAGR
GEOM Q4GS plate PMAT pmt TERM
COMP EPAI 0.01 LECT plate TERM
          0.18 LECT pmt TERM
     COUL VERT LECT plate TERM
          ROUG LECT pmt TERM
MATE LINE RO 8000.0 YOUN 2.E11 NU 0.3
          LECT tous TERM
INIT VITE 3 -100 LECT pmt TERM
OPTI PINS DUMP
          STAT
          ASN
LINK DECO BLOQ 123456 LECT c1 c2 c3 c4 TERM
          PINB PENA SFAC 1.0
               BODY LECT plate TERM
               BODY DIAM 0.18 LECT pmt TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA FDEC CONT ECRO TFRE 1.E-3
     FICH ALIC FREQ 1
OPTI NOTE
 CSTA 0.5E0
 LOG 1
 DPMA
CALC TINI 0. TEND 5.E-3
*================================================================
PLAY
CAME   1 EYE   0.00000E+00 -4.99864E+00  2.35734E+00
!        Q     8.43391E-01  5.37300E-01  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  9.06308E-01 -4.22618E-01
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  4.22618E-01  9.06308E-01
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  0.00000E+00  0.00000E+00  2.64326E-02
!RSPHERE:  1.49065E+00
!RADIUS :  5.51539E+00
!ASPECT :  1.00000E+00
!NEAR   :  4.02475E+00
!FAR    :  8.49669E+00
SCEN GEOM NAVI FREE
          PINB CDES
     COLO PAPE
     LIMA ON
SLER CAM1 1 NFRA 1
TRAC OFFS FICH AVI NOCL NFTO 557 FPS 25 KFRE 10 COMP -1 REND
FREQ 1
GOTR LOOP 555 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDPLAY
*================================================================
SUIT
Post treatment from ALIC file
ECHO
RESU ALIC GARD PSCR
SORT GRAP
AXTE 1.0 'Time [s]'
COUR 1 'dz_pmt' DEPL COMP 3 NOEU LECT pmt TERM
COUR 2 'dx_pmt' DEPL COMP 1 NOEU LECT pmt TERM
COUR 3 'vz_pmt' VITE COMP 3 NOEU LECT pmt TERM
COUR 5 'fz_pmt' FEXT COMP 3 NOEU LECT pmt TERM
RCOU 11 'dz_pmt' FICH 'pmat04.pun' RENA 'dz_pmt_04'
RCOU 13 'vz_pmt' FICH 'pmat04.pun' RENA 'vz_pmt_04'
RCOU 15 'fz_pmt' FICH 'pmat04.pun' RENA 'fz_pmt_04'
TRAC 1 AXES 1.0 'DISPL. [M]' YZER
TRAC 3 AXES 1.0 'VELOC. [M/S]' YZER
TRAC 5 AXES 1.0 'FEXT. [N]' YZER
TRAC 2 AXES 1.0 'DISPL. [M]' YZER
LIST 1 AXES 1.0 'DISPL. [M]' YZER
LIST 3 AXES 1.0 'VELOC. [M/S]' YZER
LIST 5 AXES 1.0 'FEXT. [N]' YZER
LIST 2 AXES 1.0 'DISPL. [M]' YZER
TRAC 1 11 AXES 1.0 'DISPL. [M]' YZER
COLO NOIR ROUG
TRAC 3 13 AXES 1.0 'VELOC. [M/S]' YZER
COLO NOIR ROUG
TRAC 5 15 AXES 1.0 'FEXT. [N]' YZER
COLO NOIR ROUG
*================================================================
FIN
```

pmat06.dgibi

```
opti echo 1;
opti dime 3 elem cub8;
opti sauv form 'pmat06.msh';
opti trac psc ftra 'pmat06_mesh.ps';
p1 = -1 -1 0;
p2 =  1 -1 0;
p3 =  1  1 0;
p4 = -1  1 0;
n = 19;
c1 = p1 d n p2;
c2 = p2 d n p3;
c3 = p3 d n p4;
c4 = p4 d n p1;
plate = dall c1 c2 c3 c4 plan;
pm = 0 0 0.12;
pmt = manu poi1 pm;
mesh = plate et pmt;
tass mesh;
sauv form mesh;
trac cach qual mesh;
fin;
```

pmat06.epx

```
PMAT06
ECHO
!CONV win
CAST mesh
TRID LAGR
GEOM Q4GS plate PMAT pmt TERM
COMP EPAI 0.01 LECT plate TERM
          0.18 LECT pmt TERM
     COUL VERT LECT plate TERM
          ROUG LECT pmt TERM
MATE LINE RO 8000.0 YOUN 2.E11 NU 0.3
          LECT tous TERM
INIT VITE 3 -100 LECT pmt TERM
OPTI PINS DUMP
          STAT
          ASN
LINK COUP BLOQ 123456 LECT c1 c2 c3 c4 TERM
          PINB
               BODY LECT plate TERM
               BODY DIAM 0.18 LECT pmt TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA FDEC CONT ECRO TFRE 1.E-3
     FICH ALIC FREQ 1
OPTI NOTE
 CSTA 0.5E0
 LOG 1
 DPMA
CALC TINI 0. TEND 5.E-3
*================================================================
PLAY
CAME   1 EYE   0.00000E+00 -4.99864E+00  2.35734E+00
!        Q     8.43391E-01  5.37300E-01  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  9.06308E-01 -4.22618E-01
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  4.22618E-01  9.06308E-01
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  0.00000E+00  0.00000E+00  2.64326E-02
!RSPHERE:  1.49065E+00
!RADIUS :  5.51539E+00
!ASPECT :  1.00000E+00
!NEAR   :  4.02475E+00
!FAR    :  8.49669E+00
SCEN GEOM NAVI FREE
          PINB CDES
     COLO PAPE
     LIMA ON
SLER CAM1 1 NFRA 1
TRAC OFFS FICH AVI NOCL NFTO 557 FPS 25 KFRE 10 COMP -1 REND
FREQ 1
GOTR LOOP 555 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDPLAY
*================================================================
SUIT
Post treatment from ALIC file
ECHO
RESU ALIC GARD PSCR
SORT GRAP
AXTE 1.0 'Time [s]'
COUR 1 'dz_pmt' DEPL COMP 3 NOEU LECT pmt TERM
COUR 2 'dx_pmt' DEPL COMP 1 NOEU LECT pmt TERM
COUR 3 'vz_pmt' VITE COMP 3 NOEU LECT pmt TERM
COUR 5 'fz_pmt' FLIA COMP 3 NOEU LECT pmt TERM
RCOU 11 'dz_pmt' FICH 'pmat04.pun' RENA 'dz_pmt_04'
RCOU 13 'vz_pmt' FICH 'pmat04.pun' RENA 'vz_pmt_04'
RCOU 15 'fz_pmt' FICH 'pmat04.pun' RENA 'fz_pmt_04'
TRAC 1 AXES 1.0 'DISPL. [M]' YZER
TRAC 3 AXES 1.0 'VELOC. [M/S]' YZER
TRAC 5 AXES 1.0 'FEXT. [N]' YZER
TRAC 2 AXES 1.0 'DISPL. [M]' YZER
LIST 1 AXES 1.0 'DISPL. [M]' YZER
LIST 3 AXES 1.0 'VELOC. [M/S]' YZER
LIST 5 AXES 1.0 'FEXT. [N]' YZER
LIST 2 AXES 1.0 'DISPL. [M]' YZER
TRAC 1 11 AXES 1.0 'DISPL. [M]' YZER
COLO NOIR ROUG
TRAC 3 13 AXES 1.0 'VELOC. [M/S]' YZER
COLO NOIR ROUG
TRAC 5 15 AXES 1.0 'FEXT. [N]' YZER
COLO NOIR ROUG
*================================================================
FIN
```

pmat07.dgibi

```
opti echo 1;
opti dime 3 elem cub8;
opti sauv form 'pmat07.msh';
opti trac psc ftra 'pmat07_mesh.ps';
p1 = -1 -1 0;
p2 =  1 -1 0;
p3 =  1  1 0;
p4 = -1  1 0;
n = 20;
c1 = p1 d n p2;
c2 = p2 d n p3;
c3 = p3 d n p4;
c4 = p4 d n p1;
plate = dall c1 c2 c3 c4 plan;
pm = 0 0 0.12;
pmt = manu poi1 pm;
mesh = plate et pmt;
tass mesh;
sauv form mesh;
trac cach qual mesh;
fin;
```

pmat07.epx

```
PMAT07
ECHO
!CONV win
CAST mesh
TRID LAGR
GEOM Q4GS plate PMAT pmt TERM
COMP EPAI 0.01 LECT plate TERM
          0.18 LECT pmt TERM
     COUL VERT LECT plate TERM
          ROUG LECT pmt TERM
MATE LINE RO 8000.0 YOUN 2.E11 NU 0.3
          LECT tous TERM
INIT VITE 3 -100 LECT pmt TERM
OPTI PINS DUMP
          STAT
          ASN
LINK COUP BLOQ 123456 LECT c1 c2 c3 c4 TERM
          PINB
               BODY LECT plate TERM
               BODY DIAM 0.18 LECT pmt TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA FDEC CONT ECRO TFRE 1.E-3
     FICH ALIC FREQ 1
OPTI NOTE
 CSTA 0.5E0
 LOG 1
 DPMA
CALC TINI 0. TEND 5.E-3
*=================================================================
PLAY
CAME   1 EYE   0.00000E+00 -4.99864E+00  2.35734E+00
!        Q     8.43391E-01  5.37300E-01  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  9.06308E-01 -4.22618E-01
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  4.22618E-01  9.06308E-01
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  0.00000E+00  0.00000E+00  2.64326E-02
!RSPHERE:  1.49065E+00
!RADIUS :  5.51539E+00
!ASPECT :  1.00000E+00
!NEAR   :  4.02475E+00
!FAR    :  8.49669E+00
SCEN GEOM NAVI FREE
          PINB CDES
     COLO PAPE
     LIMA ON
SLER CAM1 1 NFRA 1
TRAC OFFS FICH AVI NOCL NFTO 557 FPS 25 KFRE 10 COMP -1 REND
FREQ 1
GOTR LOOP 555 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDPLAY
*=================================================================
SUIT
Post treatment from ALIC file
ECHO
RESU ALIC GARD PSCR
SORT GRAP
AXTE 1.0 'Time [s]'
COUR 1 'dz_pmt' DEPL COMP 3 NOEU LECT pmt TERM
COUR 2 'dx_pmt' DEPL COMP 1 NOEU LECT pmt TERM
COUR 3 'vz_pmt' VITE COMP 3 NOEU LECT pmt TERM
COUR 5 'fz_pmt' FEXT COMP 3 NOEU LECT pmt TERM
RCOU 11 'dz_pmt' FICH 'pmat05.pun' RENA 'dz_pmt_05'
RCOU 13 'vz_pmt' FICH 'pmat05.pun' RENA 'vz_pmt_05'
RCOU 15 'fz_pmt' FICH 'pmat05.pun' RENA 'fz_pmt_05'
RCOU 21 'dz_pmt' FICH 'pmat06.pun' RENA 'dz_pmt_06'
RCOU 23 'vz_pmt' FICH 'pmat06.pun' RENA 'vz_pmt_06'
RCOU 25 'fz_pmt' FICH 'pmat06.pun' RENA 'fz_pmt_06'
TRAC 1 AXES 1.0 'DISPL. [M]' YZER
TRAC 3 AXES 1.0 'VELOC. [M/S]' YZER
TRAC 5 AXES 1.0 'FEXT. [N]' YZER
TRAC 2 AXES 1.0 'DISPL. [M]' YZER
LIST 1 AXES 1.0 'DISPL. [M]' YZER
LIST 3 AXES 1.0 'VELOC. [M/S]' YZER
LIST 5 AXES 1.0 'FEXT. [N]' YZER
LIST 2 AXES 1.0 'DISPL. [M]' YZER
TRAC 1 11 21 AXES 1.0 'DISPL. [M]' YZER
COLO NOIR ROUG VERT
TRAC 3 13 23 AXES 1.0 'VELOC. [M/S]' YZER
COLO NOIR ROUG VERT
TRAC 5 15 25 AXES 1.0 'FEXT. [N]' YZER
COLO NOIR ROUG VERT
*=================================================================
FIN
```

rebo01.epx

```
REBO01
ECHO
!CONV win
CPLA LAGR
GEOM LIBR POIN 12 CAR1 4 TERM
 0 0  1 0  0 1  1 1  0 2  1 2
 0 2  1 2  0 3  1 3  0 4  1 4
 1 2 4 3
 3 4 6 5
 7 8 10 9
 9 10 12 11
COMP COUL VERT LECT tous TERM
MATE LINE RO 8000.0 YOUN 1.E12 NU 0.0
          LECT tous TERM
OPTI PINS DUMP
          STAT
         !VIDE
         !EQVL
         !EQVD
          ASN
          NORB
LINK DECO PINB PENA SFAC 1.0
               BODY LECT 2 TERM
               BODY LECT 3 TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA CONT ECRO FREQ 1
     FICH ALIC FREQ 1
OPTI NOTE
 CSTA 0.5E0
 LOG 1
CALCUL TINI 0. TEND 100.E-3 NMAX 20
*=================================================================
PLAY
CAME   1 EYE   5.00000E-01  2.21000E+00  1.21286E+01
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
FREQ 2
GO
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB PARE
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB NASN PASN
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB NORM
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
FREQ 20
GO
ENDPLAY
*=================================================================
SUIT
Post-treatment (time curves from alice file)
ECHO
RESU ALIC GARD PSCR
SORT GRAP
AXTE 1.0 'Time [s]'
COUR  1 'dy_5'  DEPL COMP 2 NOEU LECT  5 TERM
COUR  2 'dy_7'  DEPL COMP 2 NOEU LECT  7 TERM
COUR 11 'fy_5'  FEXT COMP 2 NOEU LECT  5 TERM
COUR 12 'fy_7'  FEXT COMP 2 NOEU LECT  7 TERM
TRAC 1 2 AXES 1.0 'DISPL. [M]' YZER
TRAC 11 12 AXES 1.0 'FEXT [N]' YZER
LIST 1 2 AXES 1.0 'DISPL. [M]' YZER
LIST 11 12 AXES 1.0 'FEXT [N]' YZER
*=================================================================
FIN
```

rebo02.epx

```
REBO02
ECHO
!CONV win
CPLA LAGR
GEOM LIBR POIN 12 CAR1 4 TERM
 0 0  1 0  0 1  1 1  0 2  1 2
 0 2  1 2  0 3  1 3  0 4  1 4
 1 2 4 3
 3 4 6 5
 7 8 10 9
 9 10 12 11
COMP COUL VERT LECT tous TERM
MATE LINE RO 8000.0 YOUN 1.E12 NU 0.0
          LECT tous TERM
OPTI PINS DUMP
          STAT
         !VIDE
         !EQVL
         !EQVD
          ASN
          NORB
LINK DECO PINB PENA SFAC 1.0
               BODY MLEV 2 LECT 2 TERM
               BODY MLEV 2 LECT 3 TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA CONT ECRO FREQ 1
     FICH ALIC FREQ 1
OPTI NOTE
 CSTA 0.5E0
 LOG 1
CALCUL TINI 0. TEND 100.E-3 NMAX 20
*=================================================================
```

```
PLAY
CAME   1 EYE   5.00000E-01  2.21000E+00  1.21286E+01
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
FREQ 2
GO
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB PARE
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB NASN PASN
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB NORM
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
FREQ 20
GO
ENDPLAY
*==================================================================
SUIT
Post-treatment (time curves from alice file)
ECHO
RESU ALIC GARD PSCR
SORT GRAP
AXTE 1.0 'Time [s]'
COUR  1 'dy_5'  DEPL COMP 2 NOEU LECT  5 TERM
COUR  2 'dy_7'  DEPL COMP 2 NOEU LECT  7 TERM
COUR 11 'fy_5'  FEXT COMP 2 NOEU LECT  5 TERM
COUR 12 'fy_7'  FEXT COMP 2 NOEU LECT  7 TERM
TRAC 1 2 AXES 1.0 'DISPL. [M]' YZER
TRAC 11 12 AXES 1.0 'FEXT [N]' YZER
LIST 1 2 AXES 1.0 'DISPL. [M]' YZER
LIST 11 12 AXES 1.0 'FEXT [N]' YZER
*==================================================================
FIN
```

rebo03.epx

```
REBO03
ECHO
!CONV win
CPLA LAGR
GEOM LIBR POIN 12 CAR1 4 TERM
 0 0  1 0  0 1  1 1  0 2  1 2
 0 2  1 2  0 3  1 3  0 4  1 4
 1 2 4 3
 3 4 6 5
 7 8 10 9
 9 10 12 11
COMP COUL VERT LECT tous TERM
MATE LINE RO 8000.0 YOUN 1.E12 NU 0.0
          LECT tous TERM
OPTI PINS DUMP
          STAT
         !VIDE
         !EQVL
         !EQVD
         !ASN
LINK COUP PINB BODY LECT 2 TERM
               BODY LECT 3 TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA CONT ECRO FREQ 1
     FICH ALIC FREQ 1
OPTI NOTE
 CSTA 0.5E0
 LOG 1
CALCUL TINI 0. TEND 100.E-3 NMAX 20
*==================================================================
PLAY
CAME   1 EYE   5.00000E-01  2.21000E+00  1.21286E+01
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
FREQ 2
GO
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB PARE
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB NASN PASN
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB NORM
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
FREQ 20
GO
ENDPLAY
*==================================================================
SUIT
Post-treatment (time curves from alice file)
ECHO
RESU ALIC GARD PSCR
SORT GRAP
AXTE 1.0 'Time [s]'
COUR  1 'dy_5'  DEPL COMP 2 NOEU LECT  5 TERM
COUR  2 'dy_7'  DEPL COMP 2 NOEU LECT  7 TERM
COUR 11 'fy_5'  FEXT COMP 2 NOEU LECT  5 TERM
COUR 12 'fy_7'  FEXT COMP 2 NOEU LECT  7 TERM
TRAC 1 2 AXES 1.0 'DISPL. [M]' YZER
TRAC 11 12 AXES 1.0 'FEXT [N]' YZER
LIST 1 2 AXES 1.0 'DISPL. [M]' YZER
LIST 11 12 AXES 1.0 'FEXT [N]' YZER
*==================================================================
FIN
```

rebo04.epx

```
REBO04
ECHO
!CONV win
CPLA LAGR
GEOM LIBR POIN 12 CAR1 4 TERM
 0 0  1 0  0 1  1 1  0 2  1 2
 0 2  1 2  0 3  1 3  0 4  1 4
 1 2 4 3
 3 4 6 5
 7 8 10 9
 9 10 12 11
COMP COUL VERT LECT tous TERM
MATE LINE RO 8000.0 YOUN 1.E12 NU 0.0
          LECT tous TERM
OPTI PINS DUMP
          STAT
         !VIDE
         !EQVL
         !EQVD
         !ASN
LINK COUP PINB BODY MLEV 2 LECT 2 TERM
               BODY MLEV 2 LECT 3 TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA CONT ECRO FREQ 1
     FICH ALIC FREQ 1
OPTI NOTE
 CSTA 0.5E0
 LOG 1
CALCUL TINI 0. TEND 100.E-3 NMAX 20
*==================================================================
PLAY
CAME   1 EYE   5.00000E-01  2.21000E+00  1.21286E+01
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
FREQ 2
GO
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB PARE
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB NASN PASN
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB NORM
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
FREQ 20
GO
ENDPLAY
*==================================================================
SUIT
Post-treatment (time curves from alice file)
ECHO
RESU ALIC GARD PSCR
SORT GRAP
AXTE 1.0 'Time [s]'
COUR  1 'dy_5'  DEPL COMP 2 NOEU LECT  5 TERM
COUR  2 'dy_7'  DEPL COMP 2 NOEU LECT  7 TERM
COUR 11 'fy_5'  FEXT COMP 2 NOEU LECT  5 TERM
COUR 12 'fy_7'  FEXT COMP 2 NOEU LECT  7 TERM
TRAC 1 2 AXES 1.0 'DISPL. [M]' YZER
TRAC 11 12 AXES 1.0 'FEXT [N]' YZER
LIST 1 2 AXES 1.0 'DISPL. [M]' YZER
LIST 11 12 AXES 1.0 'FEXT [N]' YZER
*==================================================================
FIN
```

size01.epx

```
SIZE01
ECHO
!CONV win
CPLA LAGR
GEOM LIBR POIN 12 CAR1 4 TERM
 0 0  .1 0  0 .1  .1 .1  0 .2  .1 .2
 0 .242  .1 .242  0 .342  .1 .342  0 .442  .1 .442
 1 2 4 3
 3 4 6 5
 7 8 10 9
 9 10 12 11
COMP COUL VERT LECT tous TERM
MATE LINE RO 8000.0 YOUN 1.E12 NU 0.0
          LECT tous TERM
INIT VITE 2  50 LECT 1 PAS 1 6 TERM
     VITE 2 -50 LECT 7 PAS 1 12 TERM
OPTI PINS DUMP
          STAT
         !VIDE
         !EQVL
         !EQVD
```

```
          ASN
          NORB
LINK DECO PINB PENA SFAC 1.0
               BODY LECT 2 TERM
               BODY LECT 3 TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA CONT ECRO FREQ 1
     FICH ALIC FREQ 1
OPTI NOTE
 CSTA 0.5E0
 LOG 1
CALCUL TINI 0. TEND 100.E-3 NMAX 20
*==================================================================
PLAY
CAME   1 EYE   5.00000E-02  2.21000E-01  1.21286E+00
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
FREQ 2
GO
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB PARE
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB NASN PASN
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB NORM
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
FREQ 20
GO
ENDPLAY
*==================================================================
SUIT
Post-treatment (time curves from alice file)
ECHO
RESU ALIC GARD PSCR
SORT GRAP
AXTE 1.0 'Time [s]'
COUR  1 'dy_5'  DEPL COMP 2 NOEU LECT  5 TERM
COUR  2 'dy_7'  DEPL COMP 2 NOEU LECT  7 TERM
COUR 11 'fy_5'  FEXT COMP 2 NOEU LECT  5 TERM
COUR 12 'fy_7'  FEXT COMP 2 NOEU LECT  7 TERM
TRAC 1 2 AXES 1.0 'DISPL. [M]' YZER
TRAC 11 12 AXES 1.0 'FEXT [N]' YZER
LIST 1 2 AXES 1.0 'DISPL. [M]' YZER
LIST 11 12 AXES 1.0 'FEXT [N]' YZER
*==================================================================
FIN
```

size02.epx

```
SIZE02
ECHO
!CONV win
TRID LAGR
GEOM LIBR POIN 24 CUBE 4 TERM
 0 0 0  .1 0 0  0 .1 0  .1 .1 0  0 .2 0  .1 .2 0
 0 .274 0  .1 .274 0  0 .374 0  .1 .374 0  0 .474 0  .1 .474 0
 0 0 .1  .1 0 .1  0 .1 .1  .1 .1 .1  0 .2 .1  .1 .2 .1
 0 .274 .1  .1 .274 .1  0 .374 .1  .1 .374 .1  0 .474 .1  .1 .474 .1
 1 2 4 3 13 14 16 15
 3 4 6 5 15 16 18 17
 7 8 10 9 19 20 22 21
 9 10 12 11 21 22 24 23
COMP COUL VERT LECT tous TERM
MATE LINE RO 8000.0 YOUN 1.E12 NU 0.0
          LECT tous TERM
INIT VITE 2  50 LECT 1 PAS 1 6 13 PAS 1 18 TERM
     VITE 2 -50 LECT 7 PAS 1 12 19 PAS 1 24 TERM
OPTI PINS DUMP
          STAT
         !VIDE
         !EQVL
         !EQVD
          ASN
          NORB
LINK DECO PINB PENA SFAC 1.0
               BODY LECT 2 TERM
               BODY LECT 3 TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA CONT ECRO FREQ 1
     FICH ALIC FREQ 1
OPTI NOTE
 CSTA 0.5E0
 LOG 1
CALCUL TINI 0. TEND 100.E-3 NMAX 20
*==================================================================
PLAY
CAME   1 EYE   5.00000E-02  2.21000E-01  1.21286E+00
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
FREQ 2
GO
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB PARE
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB NASN PASN
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB NORM
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
FREQ 20
GO
ENDPLAY
*==================================================================
SUIT
Post-treatment (time curves from alice file)
ECHO
RESU ALIC GARD PSCR
SORT GRAP
AXTE 1.0 'Time [s]'
COUR  1 'dy_5'  DEPL COMP 2 NOEU LECT  5 TERM
COUR  2 'dy_7'  DEPL COMP 2 NOEU LECT  7 TERM
COUR 11 'fy_5'  FEXT COMP 2 NOEU LECT  5 TERM
COUR 12 'fy_7'  FEXT COMP 2 NOEU LECT  7 TERM
TRAC 1 2 AXES 1.0 'DISPL. [M]' YZER
TRAC 11 12 AXES 1.0 'FEXT [N]' YZER
LIST 1 2 AXES 1.0 'DISPL. [M]' YZER
LIST 11 12 AXES 1.0 'FEXT [N]' YZER
*==================================================================
FIN
```

size03.epx

```
SIZE03
ECHO
!CONV win
CPLA LAGR
GEOM LIBR POIN 12 CAR1 4 TERM
 0 0  .1 0  0 .1  .1 .1  0 .2  .1 .2
 0 .211  .1 .211  0 .311  .1 .311  0 .411  .1 .411
 1 2 4 3
 3 4 6 5
 7 8 10 9
 9 10 12 11
COMP COUL VERT LECT tous TERM
MATE LINE RO 8000.0 YOUN 1.E12 NU 0.0
          LECT tous TERM
INIT VITE 2  50 LECT 1 PAS 1 6 TERM
     VITE 2 -50 LECT 7 PAS 1 12 TERM
OPTI PINS DUMP
          STAT
         !VIDE
         !EQVL
         !EQVD
          ASN
          NORB
LINK DECO PINB PENA SFAC 1.0
               BODY MLEV 2 LECT 2 TERM
               BODY MLEV 2 LECT 3 TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA CONT ECRO FREQ 1
     FICH ALIC FREQ 1
OPTI NOTE
 CSTA 0.5E0
 LOG 1
CALCUL TINI 0. TEND 100.E-3 NMAX 20
*==================================================================
PLAY
CAME   1 EYE   5.00000E-02  2.21000E-01  1.21286E+00
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
FREQ 2
GO
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB PARE
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB NASN PASN
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB NORM
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
FREQ 20
GO
ENDPLAY
*==================================================================
SUIT
Post-treatment (time curves from alice file)
ECHO
RESU ALIC GARD PSCR
SORT GRAP
AXTE 1.0 'Time [s]'
COUR  1 'dy_5'  DEPL COMP 2 NOEU LECT  5 TERM
COUR  2 'dy_7'  DEPL COMP 2 NOEU LECT  7 TERM
COUR 11 'fy_5'  FEXT COMP 2 NOEU LECT  5 TERM
COUR 12 'fy_7'  FEXT COMP 2 NOEU LECT  7 TERM
TRAC 1 2 AXES 1.0 'DISPL. [M]' YZER
TRAC 11 12 AXES 1.0 'FEXT [N]' YZER
LIST 1 2 AXES 1.0 'DISPL. [M]' YZER
LIST 11 12 AXES 1.0 'FEXT [N]' YZER
*==================================================================
FIN
```

size04.epx

```
SIZE04
ECHO
!CONV win
TRID LAGR
GEOM LIBR POIN 24 CUBE 4 TERM
 0 0 0  .1 0 0  0 .1 0  .1 .1 0  0 .2 0  .1 .2 0
 0 .219 0  .1 .219 0  0 .319 0  .1 .319 0  0 .419 0  .1 .419 0
 0 0 .1  .1 0 .1  0 .1 .1  .1 .1 .1  0 .2 .1  .1 .2 .1
 0 .219 .1  .1 .219 .1  0 .319 .1  .1 .319 .1  0 .419 .1  .1 .419 .1
 1 2 4 3 13 14 16 15
 3 4 6 5 15 16 18 17
 7 8 10 9 19 20 22 21
 9 10 12 11 21 22 24 23
COMP COUL VERT LECT tous TERM
MATE LINE RO 8000.0 YOUN 1.E12 NU 0.0
          LECT tous TERM
INIT VITE 2  50 LECT 1 PAS 1 6 13 PAS 1 18 TERM
     VITE 2 -50 LECT 7 PAS 1 12 19 PAS 1 24 TERM
OPTI PINS DUMP
          STAT
         !VIDE
         !EQVL
         !EQVD
          ASN
          NORB
LINK DECO PINB PENA SFAC 1.0
               BODY MLEV 2 LECT 2 TERM
               BODY MLEV 2 LECT 3 TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA CONT ECRO FREQ 1
     FICH ALIC FREQ 1
OPTI NOTE
 CSTA 0.5E0
 LOG 1
CALCUL TINI 0. TEND 100.E-3 NMAX 20
*===================================================================
PLAY
CAME   1 EYE   5.00000E-02  2.21000E-01  1.21286E+00
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
FREQ 2
GO
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB PARE
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB NASN PASN
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB NORM
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
FREQ 20
GO
ENDPLAY
*===================================================================
SUIT
Post-treatment (time curves from alice file)
ECHO
RESU ALIC GARD PSCR
SORT GRAP
AXTE 1.0 'Time [s]'
COUR  1 'dy_5'  DEPL COMP 2 NOEU LECT  5 TERM
COUR  2 'dy_7'  DEPL COMP 2 NOEU LECT  7 TERM
COUR 11 'fy_5'  FEXT COMP 2 NOEU LECT  5 TERM
COUR 12 'fy_7'  FEXT COMP 2 NOEU LECT  7 TERM
TRAC 1 2 AXES 1.0 'DISPL. [M]' YZER
TRAC 11 12 AXES 1.0 'FEXT [N]' YZER
LIST 1 2 AXES 1.0 'DISPL. [M]' YZER
LIST 11 12 AXES 1.0 'FEXT [N]' YZER
*===================================================================
FIN
```

size05.epx

```
SIZE05
ECHO
!CONV win
CPLA LAGR
GEOM LIBR POIN 12 CAR1 4 TERM
 0 0  .1 0  0 .1  .1 .1  0 .2  .1 .2
 0 .206  .1 .206  0 .306  .1 .306  0 .406  .1 .406
 1 2 4 3
 3 4 6 5
 7 8 10 9
 9 10 12 11
COMP COUL VERT LECT tous TERM
MATE LINE RO 8000.0 YOUN 1.E12 NU 0.0
          LECT tous TERM
INIT VITE 2  50 LECT 1 PAS 1 6 TERM
     VITE 2 -50 LECT 7 PAS 1 12 TERM
OPTI PINS DUMP
          STAT
         !VIDE
         !EQVL
         !EQVD
          ASN
          NORB
LINK DECO PINB PENA SFAC 1.0
               BODY MLEV 3 LECT 2 TERM
               BODY MLEV 3 LECT 3 TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA CONT ECRO FREQ 1
     FICH ALIC FREQ 1
OPTI NOTE
 CSTA 0.5E0
 LOG 1
CALCUL TINI 0. TEND 100.E-3 NMAX 20
*===================================================================
PLAY
CAME   1 EYE   5.00000E-02  2.21000E-01  1.21286E+00
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
FREQ 2
GO
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB PARE
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB NASN PASN
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB NORM
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
FREQ 20
GO
ENDPLAY
*===================================================================
SUIT
Post-treatment (time curves from alice file)
ECHO
RESU ALIC GARD PSCR
SORT GRAP
AXTE 1.0 'Time [s]'
COUR  1 'dy_5'  DEPL COMP 2 NOEU LECT  5 TERM
COUR  2 'dy_7'  DEPL COMP 2 NOEU LECT  7 TERM
COUR 11 'fy_5'  FEXT COMP 2 NOEU LECT  5 TERM
COUR 12 'fy_7'  FEXT COMP 2 NOEU LECT  7 TERM
TRAC 1 2 AXES 1.0 'DISPL. [M]' YZER
TRAC 11 12 AXES 1.0 'FEXT [N]' YZER
LIST 1 2 AXES 1.0 'DISPL. [M]' YZER
LIST 11 12 AXES 1.0 'FEXT [N]' YZER
RCOU 21 'dy_5' FICH 'size01.pun' RENA 'dy_5_01'
RCOU 22 'dy_7' FICH 'size01.pun' RENA 'dy_7_01'
RCOU 31 'fy_5' FICH 'size01.pun' RENA 'fy_5_01'
RCOU 32 'fy_7' FICH 'size01.pun' RENA 'fy_7_01'
RCOU 41 'dy_5' FICH 'size03.pun' RENA 'dy_5_03'
RCOU 42 'dy_7' FICH 'size03.pun' RENA 'dy_7_03'
RCOU 51 'fy_5' FICH 'size03.pun' RENA 'fy_5_03'
RCOU 52 'fy_7' FICH 'size03.pun' RENA 'fy_7_03'
TRAC 1 2 21 22 41 42 AXES 1.0 'DISPL. [M]' YZER
COLO NOIR NOIR ROUG ROUG VERT VERT
TRAC 11 12 31 32 51 52 AXES 1.0 'FEXT [N]' YZER
COLO NOIR NOIR ROUG ROUG VERT VERT
*===================================================================
FIN
```

size06.epx

```
SIZE06
ECHO
!CONV win
TRID LAGR
GEOM LIBR POIN 24 CUBE 4 TERM
 0 0 0  .1 0 0  0 .1 0  .1 .1 0  0 .2 0  .1 .2 0
 0 .210 0  .1 .210 0  0 .310 0  .1 .310 0  0 .410 0  .1 .410 0
 0 0 .1  .1 0 .1  0 .1 .1  .1 .1 .1  0 .2 .1  .1 .2 .1
 0 .210 .1  .1 .210 .1  0 .310 .1  .1 .310 .1  0 .410 .1  .1 .410 .1
 1 2 4 3 13 14 16 15
 3 4 6 5 15 16 18 17
 7 8 10 9 19 20 22 21
 9 10 12 11 21 22 24 23
COMP COUL VERT LECT tous TERM
MATE LINE RO 8000.0 YOUN 1.E12 NU 0.0
          LECT tous TERM
INIT VITE 2  50 LECT 1 PAS 1 6 13 PAS 1 18 TERM
     VITE 2 -50 LECT 7 PAS 1 12 19 PAS 1 24 TERM
OPTI PINS DUMP
          STAT
         !VIDE
         !EQVL
         !EQVD
          ASN
          NORB
LINK DECO PINB PENA SFAC 1.0
               BODY MLEV 3 LECT 2 TERM
               BODY MLEV 3 LECT 3 TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA CONT ECRO FREQ 1
     FICH ALIC FREQ 1
OPTI NOTE
 CSTA 0.5E0
 LOG 1
CALCUL TINI 0. TEND 100.E-3 NMAX 20
*===================================================================
PLAY
CAME   1 EYE   5.00000E-02  2.21000E-01  1.21286E+00
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
FREQ 2
GO
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB PARE
     COLO PAPE
```

```
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB NASN PASN
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB NORM
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
FREQ 20
GO
ENDPLAY
*==================================================================
SUIT
Post-treatment (time curves from alice file)
ECHO
RESU ALIC GARD PSCR
SORT GRAP
AXTE 1.0 'Time [s]'
COUR  1 'dy_5'  DEPL COMP 2 NOEU LECT  5 TERM
COUR  2 'dy_7'  DEPL COMP 2 NOEU LECT  7 TERM
COUR 11 'fy_5'  FEXT COMP 2 NOEU LECT  5 TERM
COUR 12 'fy_7'  FEXT COMP 2 NOEU LECT  7 TERM
TRAC 1 2 AXES 1.0 'DISPL. [M]' YZER
TRAC 11 12 AXES 1.0 'FEXT [N]' YZER
LIST 1 2 AXES 1.0 'DISPL. [M]' YZER
LIST 11 12 AXES 1.0 'FEXT [N]' YZER
RCOU 21 'dy_5' FICH 'size02.pun' RENA 'dy_5_02'
RCOU 22 'dy_7' FICH 'size02.pun' RENA 'dy_7_02'
RCOU 31 'fy_5' FICH 'size02.pun' RENA 'fy_5_02'
RCOU 32 'fy_7' FICH 'size02.pun' RENA 'fy_7_02'
RCOU 41 'dy_5' FICH 'size04.pun' RENA 'dy_5_04'
RCOU 42 'dy_7' FICH 'size04.pun' RENA 'dy_7_04'
RCOU 51 'fy_5' FICH 'size04.pun' RENA 'fy_5_04'
RCOU 52 'fy_7' FICH 'size04.pun' RENA 'fy_7_04'
TRAC 1 2 21 22 41 42 AXES 1.0 'DISPL. [M]' YZER
COLO NOIR NOIR ROUG ROUG VERT VERT
TRAC 11 12 31 32 51 52 AXES 1.0 'FEXT [N]' YZER
COLO NOIR NOIR ROUG ROUG VERT VERT
RCOU 61 'dy_5' FICH 'size05.pun' RENA 'dy_5_05'
RCOU 62 'dy_7' FICH 'size05.pun' RENA 'dy_7_05'
RCOU 71 'fy_5' FICH 'size05.pun' RENA 'fy_5_05'
RCOU 72 'fy_7' FICH 'size05.pun' RENA 'fy_7_05'
TRAC 1 2 61 62 AXES 1.0 'DISPL. [M]' YZER
COLO NOIR NOIR TURQ TURQ
TRAC 11 12 71 72 AXES 1.0 'FEXT [N]' YZER
COLO NOIR NOIR TURQ TURQ
COUR 81 'fy_5_05_sc' MULC 71 0.1
COUR 82 'fy_7_05_sc' MULC 72 0.1
TRAC 11 12 81 82 AXES 1.0 'FEXT [N]' YZER
COLO NOIR NOIR ROSE ROSE
*==================================================================
FIN
```

slid01.dgibi

```
opti echo 1;
opti dime 2 elem qua4;
opti sauv form 'slid01.msh';
opti trac psc ftra 'slid01_mesh.ps';
p1 = 0 0;
p2 = 10 0;
p3 = 10 2;
p4 = 0 2;
c1 = p1 d 10 p2;
c2 = p2 d 2 p3;
c3 = p3 d 10 p4;
c4 = p4 d 2 p1;
base = dall c1 c2 c3 c4 plan;
p5 = 0 2.1;
p6 = 2 2.1;
p7 = 2 4.1;
p8 = 0 4.1;
c5 = p5 d 2 p6;
c6 = p6 d 2 p7;
c7 = p7 d 2 p8;
c8 = p8 d 2 p5;
slider = dall c5 c6 c7 c8 plan;
mesh = base et slider;
tass mesh;
sauv form mesh;
trac qual mesh;
fin;
```

slid01.epx

```
SLID01
ECHO
!CONV win
CAST mesh
DPLA LAGR
GEOM CAR1 base slider TERM
COMP NGRO 1 'censl' LECT slider TERM COND NEAR POIN 1.0 3.1
          COUL VERT LECT base TERM
          ROUG LECT slider TERM
MATE LINE RO 8000.0 YOUN 2.E11 NU 0.3
          LECT tous TERM
INIT VITE 1 100 LECT slider TERM
          2 -10 LECT slider TERM
OPTI PINS DUMP
          STAT
         !ASN
LINK DECO BLOQ 12 LECT c1 TERM
          PINB PENA SFAC 1.0
               BODY MLEV 4 LECT base TERM
               BODY MLEV 4 LECT slider TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA FDEC CONT ECRO TFRE 1.E-3
     FICH ALIC FREQ 1
OPTI NOTE
 CSTA 0.5E0
 LOG 1
 DPMA
CALC TINI 0. TEND 15.E-3
*==================================================================
PLAY
CAME   1 EYE   5.00000E+00  2.05000E+00  2.40204E+01
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  5.00000E+00  2.05000E+00  0.00000E+00
!RSPHERE:  5.71913E+00
!RADIUS :  2.40204E+01
!ASPECT :  1.00000E+00
!NEAR   :  1.83012E+01
!FAR    :  3.54586E+01
SCEN GEOM NAVI FREE
          PINB CDES
     COLO PAPE
     LIMA ON
SLER CAM1 1 NFRA 1
TRAC OFFS FICH AVI NOCL NFTO 175 FPS 10 KFRE 10 COMP -1 REND
FREQ 1
GOTR LOOP 173 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDPLAY
*==================================================================
SUIT
Post treatment from ALIC file
ECHO
RESU ALIC GARD PSCR
SORT GRAP
AXTE 1.0 'Time [s]'
COUR 1 'dy_csl' DEPL COMP 2 NOEU LECT censl TERM
COUR 3 'vy_csl' VITE COMP 2 NOEU LECT censl TERM
COUR 4 'vx_csl' VITE COMP 1 NOEU LECT censl TERM
COUR 5 'fy_sl' FEXT COMP 2 ZONE LECT slider TERM
COUR 6 'fx_sl' FEXT COMP 1 ZONE LECT slider TERM
TRAC 1 AXES 1.0 'DISPL. [M]' YZER
TRAC 3 4 AXES 1.0 'VELOC. [M/S]' YZER
COLO NOIR ROUG
TRAC 5 6 AXES 1.0 'FEXT. [N]' YZER
COLO NOIR ROUG
LIST 1 AXES 1.0 'DISPL. [M]' YZER
LIST 3 4 AXES 1.0 'VELOC. [M/S]' YZER
LIST 5 6 AXES 1.0 'FEXT. [N]' YZER
*==================================================================
FIN
```

slid02.dgibi

```
opti echo 1;
opti dime 2 elem qua4;
opti sauv form 'slid02.msh';
opti trac psc ftra 'slid02_mesh.ps';
p1 = 0 0;
p2 = 10 0;
p3 = 10 2;
p4 = 0 2;
c1 = p1 d 10 p2;
c2 = p2 d 2 p3;
c3 = p3 d 10 p4;
c4 = p4 d 2 p1;
base = dall c1 c2 c3 c4 plan;
p5 = 0 2.1;
p6 = 2 2.1;
p7 = 2 4.1;
p8 = 0 4.1;
c5 = p5 d 2 p6;
c6 = p6 d 2 p7;
c7 = p7 d 2 p8;
c8 = p8 d 2 p5;
slider = dall c5 c6 c7 c8 plan;
mesh = base et slider;
tass mesh;
sauv form mesh;
trac qual mesh;
fin;
```

slid02.epx

```
SLID02
ECHO
!CONV win
CAST mesh
DPLA LAGR
GEOM CAR1 base slider TERM
COMP NGRO 1 'censl' LECT slider TERM COND NEAR POIN 1.0 3.1
          COUL VERT LECT base TERM
          ROUG LECT slider TERM
MATE LINE RO 8000.0 YOUN 2.E11 NU 0.3
          LECT tous TERM
INIT VITE 1 100 LECT slider TERM
          2 -10 LECT slider TERM
OPTI PINS DUMP
          STAT
          ASN
LINK DECO BLOQ 12 LECT c1 TERM
          PINB PENA SFAC 1.0
               BODY MLEV 4 LECT base TERM
               BODY MLEV 4 LECT slider TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA FDEC CONT ECRO TFRE 1.E-3
     FICH ALIC FREQ 1
OPTI NOTE
 CSTA 0.5E0
 LOG 1
 DPMA
CALC TINI 0. TEND 15.E-3
```

```
*=================================================================
PLAY
CAME   1 EYE   5.00000E+00  2.05000E+00  2.40204E+01
!         Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  5.00000E+00  2.05000E+00  0.00000E+00
!RSPHERE:  5.71913E+00
!RADIUS :  2.40204E+01
!ASPECT :  1.00000E+00
!NEAR   :  1.83012E+01
!FAR    :  3.54586E+01
SCEN GEOM NAVI FREE
          PINB CDES
     COLO PAPE
     LIMA ON
SLER CAM1 1 NFRA 1
TRAC OFFS FICH AVI NOCL NFTO 175 FPS 10 KFRE 10 COMP -1 REND
FREQ 1
GOTR LOOP 173 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDPLAY
*=================================================================
SUIT
Post treatment from ALIC file
ECHO
RESU ALIC GARD PSCR
SORT GRAP
AXTE 1.0 'Time [s]'
COUR 1 'dy_csl' DEPL COMP 2 NOEU LECT censl TERM
COUR 3 'vy_csl' VITE COMP 2 NOEU LECT censl TERM
COUR 4 'vx_csl' VITE COMP 1 NOEU LECT censl TERM
COUR 5 'fy_sl' FEXT COMP 2 ZONE LECT slider TERM
COUR 6 'fx_sl' FEXT COMP 1 ZONE LECT slider TERM
RCOU 14 'vx_csl' FICH 'slid01.pun' RENA 'vx_csl_01'
RCOU 16 'fx_sl' FICH 'slid01.pun' RENA 'fx_sl_01'
TRAC 1 AXES 1.0 'DISPL. [M]' YZER
TRAC 3 4 AXES 1.0 'VELOC. [M/S]' YZER
COLO NOIR ROUG
TRAC 5 6 AXES 1.0 'FEXT. [N]' YZER
COLO NOIR ROUG
LIST 1 AXES 1.0 'DISPL. [M]' YZER
LIST 3 4 AXES 1.0 'VELOC. [M/S]' YZER
LIST 5 6 AXES 1.0 'FEXT. [N]' YZER
TRAC 4 14 AXES 1.0 'VELOC. [M/S]' YZER
COLO NOIR ROUG
TRAC 6 16 AXES 1.0 'FEXT. [N]' YZER
COLO NOIR ROUG
*=================================================================
FIN
```

slid03.dgibi

```
opti echo 1;
opti dime 2 elem qua4;
opti sauv form 'slid03.msh';
opti trac psc ftra 'slid03_mesh.ps';
p1 = 0 0;
p2 = 10 0;
p3 = 10 2;
p4 = 0 2;
c1 = p1 d 10 p2;
c2 = p2 d 2 p3;
c3 = p3 d 10 p4;
c4 = p4 d 2 p1;
base = dall c1 c2 c3 c4 plan;
p5 = 0 2.1;
p6 = 2 2.1;
p7 = 2 4.1;
p8 = 0 4.1;
c5 = p5 d 2 p6;
c6 = p6 d 2 p7;
c7 = p7 d 2 p8;
c8 = p8 d 2 p5;
slider = dall c5 c6 c7 c8 plan;
mesh = base et slider;
tass mesh;
sauv form mesh;
trac qual mesh;
fin;
```

slid03.epx

```
SLID03
ECHO
!CONV win
CAST mesh
DPLA LAGR
GEOM CAR1 base slider TERM
COMP NGRO 1 'censl' LECT slider TERM COND NEAR POIN 1.0 3.1
          COUL VERT LECT base TERM
          ROUG LECT slider TERM
MATE LINE RO 8000.0 YOUN 2.E11 NU 0.3
          LECT tous TERM
INIT VITE 1 100 LECT slider TERM
          2 -10 LECT slider TERM
OPTI PINS DUMP
          STAT
         !ASN
LINK COUP BLOQ 12 LECT c1 TERM
          PINB
               BODY MLEV 4 LECT base TERM
               BODY MLEV 4 LECT slider TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA FDEC CONT ECRO TFRE 1.E-3
     FICH ALIC FREQ 1
OPTI NOTE
 CSTA 0.5E0
 LOG 1
 DPMA
CALC TINI 0. TEND 15.E-3
*=================================================================
PLAY
CAME   1 EYE   5.00000E+00  2.05000E+00  2.40204E+01
!         Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  5.00000E+00  2.05000E+00  0.00000E+00
!RSPHERE:  5.71913E+00
!RADIUS :  2.40204E+01
!ASPECT :  1.00000E+00
!NEAR   :  1.83012E+01
!FAR    :  3.54586E+01
SCEN GEOM NAVI FREE
          PINB CDES
     COLO PAPE
     LIMA ON
SLER CAM1 1 NFRA 1
TRAC OFFS FICH AVI NOCL NFTO 175 FPS 10 KFRE 10 COMP -1 REND
FREQ 1
GOTR LOOP 173 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDPLAY
*=================================================================
SUIT
Post treatment from ALIC file
ECHO
RESU ALIC GARD PSCR
SORT GRAP
AXTE 1.0 'Time [s]'
COUR 1 'dy_csl' DEPL COMP 2 NOEU LECT censl TERM
COUR 3 'vy_csl' VITE COMP 2 NOEU LECT censl TERM
COUR 4 'vx_csl' VITE COMP 1 NOEU LECT censl TERM
COUR 5 'fy_sl' FLIA COMP 2 ZONE LECT slider TERM
COUR 6 'fx_sl' FLIA COMP 1 ZONE LECT slider TERM
TRAC 1 AXES 1.0 'DISPL. [M]' YZER
TRAC 3 4 AXES 1.0 'VELOC. [M/S]' YZER
COLO NOIR ROUG
TRAC 5 6 AXES 1.0 'FLIA. [N]' YZER
COLO NOIR ROUG
LIST 1 AXES 1.0 'DISPL. [M]' YZER
LIST 3 4 AXES 1.0 'VELOC. [M/S]' YZER
LIST 5 6 AXES 1.0 'FLIA. [N]' YZER
*=================================================================
FIN
```

slid04.dgibi

```
opti echo 1;
opti dime 2 elem qua4;
opti sauv form 'slid04.msh';
opti trac psc ftra 'slid04_mesh.ps';
p1 = 0 0;
p2 = 10 0;
p3 = 10 2;
p4 = 0 2;
c1 = p1 d 10 p2;
c2 = p2 d 2 p3;
c3 = p3 d 10 p4;
c4 = p4 d 2 p1;
base = dall c1 c2 c3 c4 plan;
p5 = 0 2.1;
p6 = 2 2.1;
p7 = 2 4.1;
p8 = 0 4.1;
c5 = p5 d 2 p6;
c6 = p6 d 2 p7;
c7 = p7 d 2 p8;
c8 = p8 d 2 p5;
slider = dall c5 c6 c7 c8 plan;
mesh = base et slider;
tass mesh;
sauv form mesh;
trac qual mesh;
fin;
```

slid04.epx

```
SLID04
ECHO
!CONV win
CAST mesh
DPLA LAGR
GEOM CAR1 base slider TERM
COMP NGRO 1 'censl' LECT slider TERM COND NEAR POIN 1.0 3.1
          COUL VERT LECT base TERM
          ROUG LECT slider TERM
MATE LINE RO 8000.0 YOUN 2.E11 NU 0.3
          LECT tous TERM
INIT VITE 1 100 LECT slider TERM
          2 -10 LECT slider TERM
OPTI PINS DUMP
          STAT
          ASN
LINK COUP BLOQ 12 LECT c1 TERM
          PINB
               BODY MLEV 4 LECT base TERM
               BODY MLEV 4 LECT slider TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA FDEC CONT ECRO TFRE 1.E-3
                                                            !FREQ 1
     FICH ALIC FREQ 1
OPTI NOTE
 CSTA 0.5E0
 LOG 1
 DPMA
!DUMP
!PINS STAT DUMP
!LNKS STAT DUMP
CALC TINI 0. TEND 15.E-3 !NMAX 93
```

```
!fin
*=================================================================
PLAY
CAME   1 EYE   5.00000E+00  2.05000E+00  2.40204E+01
!        Q      1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  5.00000E+00  2.05000E+00  0.00000E+00
!RSPHERE:  5.71913E+00
!RADIUS :  2.40204E+01
!ASPECT :  1.00000E+00
!NEAR   :  1.83012E+01
!FAR    :  3.54586E+01
SCEN GEOM NAVI FREE
          PINB CDES
     COLO PAPE
     LIMA ON
SLER CAM1 1 NFRA 1
TRAC OFFS FICH AVI NOCL NFTO 175 FPS 10 KFRE 10 COMP -1 REND
FREQ 1
GOTR LOOP 173 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDPLAY
*=================================================================
SUIT
Post treatment from ALIC file
ECHO
RESU ALIC GARD PSCR
SORT GRAP
AXTE 1.0 'Time [s]'
COUR 1 'dy_csl' DEPL COMP 2 NOEU LECT censl TERM
COUR 3 'vy_csl' VITE COMP 2 NOEU LECT censl TERM
COUR 4 'vx_csl' VITE COMP 1 NOEU LECT censl TERM
COUR 5 'fy_sl' FLIA COMP 2 ZONE LECT slider TERM
COUR 6 'fx_sl' FLIA COMP 1 ZONE LECT slider TERM
RCOU 14 'vx_csl' FICH 'slid03.pun' RENA 'vx_csl_03'
RCOU 16 'fx_sl' FICH 'slid03.pun' RENA 'fx_sl_03'
TRAC 1 AXES 1.0 'DISPL. [M]' YZER
TRAC 3 4 AXES 1.0 'VELOC. [M/S]' YZER
COLO NOIR ROUG
TRAC 5 6 AXES 1.0 'FLIA. [N]' YZER
COLO NOIR ROUG
LIST 1 AXES 1.0 'DISPL. [M]' YZER
LIST 3 4 AXES 1.0 'VELOC. [M/S]' YZER
LIST 5 6 AXES 1.0 'FLIA. [N]' YZER
TRAC 4 14 AXES 1.0 'VELOC. [M/S]' YZER
COLO NOIR ROUG
TRAC 6 16 AXES 1.0 'FLIA. [N]' YZER
COLO NOIR ROUG
*=================================================================
FIN
```

slid05.epx

```
SLID05
ECHO
!CONV win
DPLA LAGR
GEOM LIBR POIN 8 CAR1 2 TERM
 0 -1 1 -1
 0  0 1  0
 0  0 1  0
 0  1 1  1
 1 2 4 3
 5 6 8 7
COMP COUL VERT LECT 1 TERM
          ROUG LECT 2 TERM
MATE LINE RO 8000.0 YOUN 2.E11 NU 0.3
          LECT tous TERM
INIT VITE 1 100 LECT 5 PAS 1 8 TERM
          2 -10 LECT 5 PAS 1 8 TERM
OPTI PINS DUMP
          STAT
         !ASN
LINK COUP PINB BODY MLEV 4 LECT 1 TERM
               BODY MLEV 4 LECT 2 TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA FDEC CONT ECRO FREQ 1
     FICH ALIC FREQ 1
OPTI NOTE
 CSTA 0.5E0
 LOG 1
 DPMA
CALC TINI 0. TEND 15.E-3 NMAX 10
*=================================================================
PLAY
CAME   1 EYE   5.00000E-01  0.00000E+00  5.64306E+00
!        Q      1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  5.00000E-01  0.00000E+00  0.00000E+00
!RSPHERE:  1.56752E+00
!RADIUS :  5.64306E+00
!ASPECT :  1.00000E+00
!NEAR   :  4.07554E+00
!FAR    :  8.77809E+00
SCEN GEOM NAVI FREE
          PINB CDES
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH AVI NOCL NFTO 11 FPS 5 KFRE 5 COMP -1 REND
FREQ 1
GOTR LOOP 9 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDPLAY
*=================================================================
SUIT
Post-treatment from Alice file
ECHO
RESU ALIC GARD PSCR
OPTI PRIN
SORT VISU NSTO 1
*=================================================================
PLAY
CAME   1 EYE   5.00000E-01  0.00000E+00  5.64306E+00
!        Q      1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  5.00000E-01  0.00000E+00  0.00000E+00
!RSPHERE:  1.56752E+00
!RADIUS :  5.64306E+00
!ASPECT :  1.00000E+00
!NEAR   :  4.07554E+00
!FAR    :  8.77809E+00
SCEN GEOM NAVI FREE
          FACE HFRO
     VECT SCCO FIEL FLIA SCAL A14
     TEXT VSCA
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH AVI NOCL NFTO 11 FPS 5 KFRE 5 COMP -1 REND
FREQ 1
GOTR LOOP 9 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDPLAY
*=================================================================
FIN
```

slid06.epx

```
SLID06
ECHO
 CONV win
DPLA LAGR
GEOM LIBR POIN 8 CAR1 2 TERM
 0 -1 1 -1
 0  0 1  0
 0  0 1  0
 0  1 1  1
 1 2 4 3
 5 6 8 7
COMP COUL VERT LECT 1 TERM
          ROUG LECT 2 TERM
MATE LINE RO 8000.0 YOUN 2.E11 NU 0.3
          LECT tous TERM
INIT VITE 1 100 LECT 5 PAS 1 8 TERM
          2 -10 LECT 5 PAS 1 8 TERM
OPTI PINS DUMP
          STAT
          ASN
LINK COUP PINB BODY MLEV 4 LECT 1 TERM
               BODY MLEV 4 LECT 2 TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA FDEC CONT ECRO FREQ 1
     FICH ALIC FREQ 1
OPTI NOTE
 CSTA 0.5E0
 LOG 1
 DPMA
 DUMP
 PINS STAT DUMP
 LNKS STAT DUMP
CALC TINI 0. TEND 15.E-3 NMAX 10
*=================================================================
PLAY
CAME   1 EYE   5.00000E-01  0.00000E+00  1.88102E+00
!        Q      1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  5.00000E-01  0.00000E+00  0.00000E+00
!RSPHERE:  1.56752E+00
!RADIUS :  1.88102E+00
!ASPECT :  1.33333E+00
!NEAR   :  3.13503E-01
!FAR    :  5.01605E+00
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB CDES JOIN
     VECT SCCO FIEL FLIA SCAL A14
     TEXT VSCA PCON
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS SIZE 800 600 FICH BMP REND
FREQ 1
GOTR LOOP 9 OFFS SIZE 800 600 FICH BMP REND
GO
TRAC OFFS SIZE 800 600 FICH BMP REND
ENDPLAY
*=================================================================
SUIT
Post-treatment from Alice file
ECHO
RESU ALIC GARD PSCR
OPTI PRIN
SORT VISU NSTO 1
*=================================================================
PLAY
CAME   1 EYE   5.00000E-01  0.00000E+00  1.88102E+00
!        Q      1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  5.00000E-01  0.00000E+00  0.00000E+00
!RSPHERE:  1.56752E+00
!RADIUS :  1.88102E+00
!ASPECT :  1.33333E+00
!NEAR   :  3.13503E-01
```

```
!FAR    :  5.01605E+00
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB CDES JOIN
     VECT SCCO FIEL VITE SCAL A14
     TEXT VSCA PCON
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS SIZE 800 600 FICH BMP REND
FREQ 1
GOTR LOOP 9 OFFS SIZE 800 600 FICH BMP REND
GO
TRAC OFFS SIZE 800 600 FICH BMP REND
ENDPLAY
*=================================================================
SUIT
Post-treatment from Alice file
ECHO
RESU ALIC GARD PSCR
OPTI PRIN
SORT VISU NSTO 1
*=================================================================
PLAY
CAME   1 EYE   5.00000E-01  0.00000E+00  5.64306E+00
!        Q      1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  5.00000E-01  0.00000E+00  0.00000E+00
!RSPHERE:  1.56752E+00
!RADIUS :  5.64306E+00
!ASPECT :  1.00000E+00
!NEAR   :  4.07554E+00
!FAR    :  8.77809E+00
SCEN GEOM NAVI FREE
          PINB CDES
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH AVI NOCL NFTO 11 FPS 5 KFRE 5 COMP -1 REND
FREQ 1
GOTR LOOP 9 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDPLAY
*=================================================================
SUIT
Post-treatment from Alice file
ECHO
RESU ALIC GARD PSCR
OPTI PRIN
SORT VISU NSTO 1
*=================================================================
PLAY
CAME   1 EYE   5.00000E-01  0.00000E+00  5.64306E+00
!        Q      1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  5.00000E-01  0.00000E+00  0.00000E+00
!RSPHERE:  1.56752E+00
!RADIUS :  5.64306E+00
!ASPECT :  1.00000E+00
!NEAR   :  4.07554E+00
!FAR    :  8.77809E+00
SCEN GEOM NAVI FREE
          FACE HFRO
     VECT SCCO FIEL FLIA SCAL A14
     TEXT VSCA
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH AVI NOCL NFTO 11 FPS 5 KFRE 5 COMP -1 REND
FREQ 1
GOTR LOOP 9 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDPLAY
*=================================================================
FIN
```

slid07.epx

```
SLID07
ECHO
 CONV win
DPLA LAGR
GEOM LIBR POIN 8 CAR1 2 TERM
 0 -1 1 -1
 0  0 1  0
 0  0 1  0
 0  1 1  1
 1 2 4 3
 5 6 8 7
COMP COUL VERT LECT 1 TERM
          ROUG LECT 2 TERM
MATE LINE RO 8000.0 YOUN 2.E11 NU 0.3
          LECT tous TERM
INIT VITE 1 100 LECT 5 PAS 1 8 TERM
          2 -10 LECT 5 PAS 1 8 TERM
OPTI PINS DUMP
          STAT
          ASN
LINK COUP PINB BODY MLEV 3 LECT 1 TERM
               BODY MLEV 3 LECT 2 TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA FDEC CONT ECRO FREQ 1
     FICH ALIC FREQ 1
OPTI NOTE
 CSTA 0.5E0
 LOG 1
 DPMA
 DUMP
 PINS STAT DUMP
 LNKS STAT DUMP
CALC TINI 0. TEND 15.E-3 NMAX 10
*=================================================================
PLAY
CAME   1 EYE   5.00000E-01  0.00000E+00  1.88102E+00
!        Q      1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  5.00000E-01  0.00000E+00  0.00000E+00
!RSPHERE:  1.56752E+00
!RADIUS :  1.88102E+00
!ASPECT :  1.33333E+00
!NEAR   :  3.13503E-01
!FAR    :  5.01605E+00
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB CDES JOIN
     VECT SCCO FIEL FLIA SCAL A14
     TEXT VSCA PCON
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS SIZE 800 600 FICH BMP REND
FREQ 1
GOTR LOOP 9 OFFS SIZE 800 600 FICH BMP REND
GO
TRAC OFFS SIZE 800 600 FICH BMP REND
ENDPLAY
*=================================================================
SUIT
Post-treatment from Alice file
ECHO
RESU ALIC GARD PSCR
OPTI PRIN
SORT VISU NSTO 1
*=================================================================
PLAY
CAME   1 EYE   5.00000E-01  0.00000E+00  1.88102E+00
!        Q      1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  5.00000E-01  0.00000E+00  0.00000E+00
!RSPHERE:  1.56752E+00
!RADIUS :  1.88102E+00
!ASPECT :  1.33333E+00
!NEAR   :  3.13503E-01
!FAR    :  5.01605E+00
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB CDES JOIN
     VECT SCCO FIEL VITE SCAL A14
     TEXT VSCA PCON
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS SIZE 800 600 FICH BMP REND
FREQ 1
GOTR LOOP 9 OFFS SIZE 800 600 FICH BMP REND
GO
TRAC OFFS SIZE 800 600 FICH BMP REND
ENDPLAY
*=================================================================
SUIT
Post-treatment from Alice file
ECHO
RESU ALIC GARD PSCR
OPTI PRIN
SORT VISU NSTO 1
*=================================================================
PLAY
CAME   1 EYE   5.00000E-01  0.00000E+00  5.64306E+00
!        Q      1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  5.00000E-01  0.00000E+00  0.00000E+00
!RSPHERE:  1.56752E+00
!RADIUS :  5.64306E+00
!ASPECT :  1.00000E+00
!NEAR   :  4.07554E+00
!FAR    :  8.77809E+00
SCEN GEOM NAVI FREE
          PINB CDES
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH AVI NOCL NFTO 11 FPS 5 KFRE 5 COMP -1 REND
FREQ 1
GOTR LOOP 9 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDPLAY
*=================================================================
SUIT
Post-treatment from Alice file
ECHO
RESU ALIC GARD PSCR
OPTI PRIN
SORT VISU NSTO 1
*=================================================================
PLAY
CAME   1 EYE   5.00000E-01  0.00000E+00  5.64306E+00
!        Q      1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  5.00000E-01  0.00000E+00  0.00000E+00
!RSPHERE:  1.56752E+00
!RADIUS :  5.64306E+00
!ASPECT :  1.00000E+00
!NEAR   :  4.07554E+00
!FAR    :  8.77809E+00
```

```
SCEN GEOM NAVI FREE
          FACE HFRO
     VECT SCCO FIEL FLIA SCAL A14
     TEXT VSCA
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH AVI NOCL NFTO 11 FPS 5 KFRE 5 COMP -1 REND
FREQ 1
GOTR LOOP 9 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDPLAY
*===================================================================
FIN
```

slid08.epx

```
SLID08
ECHO
 CONV win
DPLA LAGR
GEOM LIBR POIN 8 CAR1 2 TERM
 0 -1 1 -1
 0  0 1  0
 0  0 1  0
 0  1 1  1
 1 2 4 3
 5 6 8 7
COMP COUL VERT LECT 1 TERM
          ROUG LECT 2 TERM
MATE LINE RO 8000.0 YOUN 2.E11 NU 0.3
          LECT tous TERM
INIT VITE 1 100 LECT 5 PAS 1 8 TERM
          2 -10 LECT 5 PAS 1 8 TERM
OPTI PINS DUMP
          STAT
          ASN
LINK COUP PINB BODY MLEV 2 LECT 1 TERM
               BODY MLEV 2 LECT 2 TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA FDEC CONT ECRO FREQ 1
     FICH ALIC FREQ 1
OPTI NOTE
 CSTA 0.5E0
 LOG 1
 DPMA
 DUMP
 PINS STAT DUMP
 LNKS STAT DUMP
CALC TINI 0. TEND 15.E-3 NMAX 10
*===================================================================
PLAY
CAME   1 EYE   5.00000E-01  0.00000E+00  1.88102E+00
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  5.00000E-01  0.00000E+00  0.00000E+00
!RSPHERE:  1.56752E+00
!RADIUS :  1.88102E+00
!ASPECT :  1.33333E+00
!NEAR   :  3.13503E-01
!FAR    :  5.01605E+00
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB CDES JOIN
     VECT SCCO FIEL FLIA SCAL A14
     TEXT VSCA PCON
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS SIZE 800 600 FICH BMP REND
FREQ 1
GOTR LOOP 9 OFFS SIZE 800 600 FICH BMP REND
GO
TRAC OFFS SIZE 800 600 FICH BMP REND
ENDPLAY
*===================================================================
SUIT
Post-treatment from Alice file
ECHO
RESU ALIC GARD PSCR
OPTI PRIN
SORT VISU NSTO 1
*===================================================================
PLAY
CAME   1 EYE   5.00000E-01  0.00000E+00  1.88102E+00
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  5.00000E-01  0.00000E+00  0.00000E+00
!RSPHERE:  1.56752E+00
!RADIUS :  1.88102E+00
!ASPECT :  1.33333E+00
!NEAR   :  3.13503E-01
!FAR    :  5.01605E+00
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB CDES JOIN
     VECT SCCO FIEL VITE SCAL A14
     TEXT VSCA PCON
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS SIZE 800 600 FICH BMP REND
FREQ 1
GOTR LOOP 9 OFFS SIZE 800 600 FICH BMP REND
GO
TRAC OFFS SIZE 800 600 FICH BMP REND
ENDPLAY
*===================================================================
SUIT
Post-treatment from Alice file
ECHO
RESU ALIC GARD PSCR
OPTI PRIN
SORT VISU NSTO 1
*===================================================================
PLAY
CAME   1 EYE   5.00000E-01  0.00000E+00  5.64306E+00
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  5.00000E-01  0.00000E+00  0.00000E+00
!RSPHERE:  1.56752E+00
!RADIUS :  5.64306E+00
!ASPECT :  1.00000E+00
!NEAR   :  4.07554E+00
!FAR    :  8.77809E+00
SCEN GEOM NAVI FREE
          PINB CDES
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH AVI NOCL NFTO 11 FPS 5 KFRE 5 COMP -1 REND
FREQ 1
GOTR LOOP 9 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDPLAY
*===================================================================
SUIT
Post-treatment from Alice file
ECHO
RESU ALIC GARD PSCR
OPTI PRIN
SORT VISU NSTO 1
*===================================================================
PLAY
CAME   1 EYE   5.00000E-01  0.00000E+00  5.64306E+00
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  5.00000E-01  0.00000E+00  0.00000E+00
!RSPHERE:  1.56752E+00
!RADIUS :  5.64306E+00
!ASPECT :  1.00000E+00
!NEAR   :  4.07554E+00
!FAR    :  8.77809E+00
SCEN GEOM NAVI FREE
          FACE HFRO
     VECT SCCO FIEL FLIA SCAL A14
     TEXT VSCA
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH AVI NOCL NFTO 11 FPS 5 KFRE 5 COMP -1 REND
FREQ 1
GOTR LOOP 9 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDPLAY
*===================================================================
FIN
```

slid09.epx

```
SLID09
ECHO
 CONV win
DPLA LAGR
GEOM LIBR POIN 8 CAR1 2 TERM
 0 -1 1 -1
 0  0 1  0
 0  0 1  0
 0  1 1  1
 1 2 4 3
 5 6 8 7
COMP COUL VERT LECT 1 TERM
          ROUG LECT 2 TERM
MATE LINE RO 8000.0 YOUN 2.E11 NU 0.3
          LECT tous TERM
INIT VITE 1 100 LECT 5 PAS 1 8 TERM
          2 -10 LECT 5 PAS 1 8 TERM
OPTI PINS DUMP
          STAT
          ASN
LINK COUP PINB BODY MLEV 1 LECT 1 TERM
               BODY MLEV 1 LECT 2 TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA FDEC CONT ECRO FREQ 1
     FICH ALIC FREQ 1
OPTI NOTE
 CSTA 0.5E0
 LOG 1
 DPMA
 DUMP
 PINS STAT DUMP
 LNKS STAT DUMP
CALC TINI 0. TEND 15.E-3 NMAX 10
*===================================================================
PLAY
CAME   1 EYE   5.00000E-01  0.00000E+00  1.88102E+00
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  5.00000E-01  0.00000E+00  0.00000E+00
!RSPHERE:  1.56752E+00
!RADIUS :  1.88102E+00
!ASPECT :  1.33333E+00
!NEAR   :  3.13503E-01
!FAR    :  5.01605E+00
SCEN GEOM NAVI FREE
          FACE HFRO
```

```
        PINB CDES JOIN
     VECT SCCO FIEL FLIA SCAL A14
     TEXT VSCA PCON
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS SIZE 800 600 FICH BMP REND
FREQ 1
GOTR LOOP 9 OFFS SIZE 800 600 FICH BMP REND
GO
TRAC OFFS SIZE 800 600 FICH BMP REND
ENDPLAY
*=================================================================
SUIT
Post-treatment from Alice file
ECHO
RESU ALIC GARD PSCR
OPTI PRIN
SORT VISU NSTO 1
*=================================================================
PLAY
CAME   1 EYE   5.00000E-01  0.00000E+00  1.88102E+00
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  5.00000E-01  0.00000E+00  0.00000E+00
!RSPHERE:  1.56752E+00
!RADIUS :  1.88102E+00
!ASPECT :  1.33333E+00
!NEAR   :  3.13503E-01
!FAR    :  5.01605E+00
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB CDES JOIN
     VECT SCCO FIEL VITE SCAL A14
     TEXT VSCA PCON
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS SIZE 800 600 FICH BMP REND
FREQ 1
GOTR LOOP 9 OFFS SIZE 800 600 FICH BMP REND
GO
TRAC OFFS SIZE 800 600 FICH BMP REND
ENDPLAY
*=================================================================
SUIT
Post-treatment from Alice file
ECHO
RESU ALIC GARD PSCR
OPTI PRIN
SORT VISU NSTO 1
*=================================================================
PLAY
CAME   1 EYE   5.00000E-01  0.00000E+00  5.64306E+00
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  5.00000E-01  0.00000E+00  0.00000E+00
!RSPHERE:  1.56752E+00
!RADIUS :  5.64306E+00
!ASPECT :  1.00000E+00
!NEAR   :  4.07554E+00
!FAR    :  8.77809E+00
SCEN GEOM NAVI FREE
          PINB CDES
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH AVI NOCL NFTO 11 FPS 5 KFRE 5 COMP -1 REND
FREQ 1
GOTR LOOP 9 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDPLAY
*=================================================================
SUIT
Post-treatment from Alice file
ECHO
RESU ALIC GARD PSCR
OPTI PRIN
SORT VISU NSTO 1
*=================================================================
PLAY
CAME   1 EYE   5.00000E-01  0.00000E+00  5.64306E+00
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  5.00000E-01  0.00000E+00  0.00000E+00
!RSPHERE:  1.56752E+00
!RADIUS :  5.64306E+00
!ASPECT :  1.00000E+00
!NEAR   :  4.07554E+00
!FAR    :  8.77809E+00
SCEN GEOM NAVI FREE
          FACE HFRO
     VECT SCCO FIEL FLIA SCAL A14
     TEXT VSCA
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH AVI NOCL NFTO 11 FPS 5 KFRE 5 COMP -1 REND
FREQ 1
GOTR LOOP 9 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDPLAY
*=================================================================
FIN
```

slid10.epx

```
SLID10
ECHO
 CONV win
DPLA LAGR
GEOM LIBR POIN 8 CAR1 2 TERM
 0 -1 1 -1
 0  0 1  0
 0  0 1  0
 0  1 1  1
 1 2 4 3
 5 6 8 7
COMP COUL VERT LECT 1 TERM
          ROUG LECT 2 TERM
MATE LINE RO 8000.0 YOUN 2.E11 NU 0.3
          LECT tous TERM
INIT VITE 1 100 LECT 5 PAS 1 8 TERM
          2 -10 LECT 5 PAS 1 8 TERM
OPTI PINS DUMP
          STAT
          ASN
LINK COUP PINB BODY MLEV 0 LECT 1 TERM
               BODY MLEV 0 LECT 2 TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA FDEC CONT ECRO FREQ 1
     FICH ALIC FREQ 1
OPTI NOTE
 CSTA 0.5E0
 LOG 1
 DPMA
 DUMP
 PINS STAT DUMP
 LNKS STAT DUMP
CALC TINI 0. TEND 15.E-3 NMAX 10
*=================================================================
PLAY
CAME   1 EYE   5.00000E-01  0.00000E+00  1.88102E+00
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  5.00000E-01  0.00000E+00  0.00000E+00
!RSPHERE:  1.56752E+00
!RADIUS :  1.88102E+00
!ASPECT :  1.33333E+00
!NEAR   :  3.13503E-01
!FAR    :  5.01605E+00
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB CDES JOIN
     VECT SCCO FIEL FLIA SCAL A14
     TEXT VSCA PCON
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS SIZE 800 600 FICH BMP REND
FREQ 1
GOTR LOOP 9 OFFS SIZE 800 600 FICH BMP REND
GO
TRAC OFFS SIZE 800 600 FICH BMP REND
ENDPLAY
*=================================================================
SUIT
Post-treatment from Alice file
ECHO
RESU ALIC GARD PSCR
OPTI PRIN
SORT VISU NSTO 1
*=================================================================
PLAY
CAME   1 EYE   5.00000E-01  0.00000E+00  1.88102E+00
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  5.00000E-01  0.00000E+00  0.00000E+00
!RSPHERE:  1.56752E+00
!RADIUS :  1.88102E+00
!ASPECT :  1.33333E+00
!NEAR   :  3.13503E-01
!FAR    :  5.01605E+00
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB CDES JOIN
     VECT SCCO FIEL VITE SCAL A14
     TEXT VSCA PCON
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS SIZE 800 600 FICH BMP REND
FREQ 1
GOTR LOOP 9 OFFS SIZE 800 600 FICH BMP REND
GO
TRAC OFFS SIZE 800 600 FICH BMP REND
ENDPLAY
*=================================================================
SUIT
Post-treatment from Alice file
ECHO
RESU ALIC GARD PSCR
OPTI PRIN
SORT VISU NSTO 1
*=================================================================
PLAY
CAME   1 EYE   5.00000E-01  0.00000E+00  5.64306E+00
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  5.00000E-01  0.00000E+00  0.00000E+00
!RSPHERE:  1.56752E+00
!RADIUS :  5.64306E+00
```

```
!ASPECT :  1.00000E+00
!NEAR   :  4.07554E+00
!FAR    :  8.77809E+00
SCEN GEOM NAVI FREE
          PINB CDES
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH AVI NOCL NFTO 11 FPS 5 KFRE 5 COMP -1 REND
FREQ 1
GOTR LOOP 9 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDPLAY
*=================================================================
SUIT
Post-treatment from Alice file
ECHO
RESU ALIC GARD PSCR
OPTI PRIN
SORT VISU NSTO 1
*=================================================================
PLAY
CAME   1 EYE   5.00000E-01  0.00000E+00  5.64306E+00
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  5.00000E-01  0.00000E+00  0.00000E+00
!RSPHERE:  1.56752E+00
!RADIUS :  5.64306E+00
!ASPECT :  1.00000E+00
!NEAR   :  4.07554E+00
!FAR    :  8.77809E+00
SCEN GEOM NAVI FREE
          FACE HFRO
     VECT SCCO FIEL FLIA SCAL A14
     TEXT VSCA
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH AVI NOCL NFTO 11 FPS 5 KFRE 5 COMP -1 REND
FREQ 1
GOTR LOOP 9 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDPLAY
*=================================================================
FIN
```

vide01.epx

```
VIDE01
ECHO
!CONV win
CPLA LAGR
GEOM LIBR POIN 4 CAR1 1 TERM
 0 0  1 0  1 1  0 1
 1 2 3 4
COMP COUL VERT LECT tous TERM
MATE LINE RO 8000.0 YOUN 1.E12 NU 0.0
          LECT tous TERM
OPTI PINS DUMP
          STAT
          VIDE
         !EQVL
         !EQVD
          ASN
LINK COUP
     PINB BODY MLEV 2 LECT tous TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA CONT ECRO FREQ 1
     FICH ALIC FREQ 1
OPTI NOTE
 CSTA 0.5E0
 LOG 1
CALCUL TINI 0. TEND 100.E-3
*=================================================================
PLAY
CAME   1 EYE   5.00000E-01  5.00000E-01  6.12372E+00
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB PARE NASN PASN
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB CDES DASN
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
ENDPLAY
*=================================================================
FIN
```

vide02.epx

```
VIDE02
ECHO
!CONV win
CPLA LAGR
GEOM LIBR POIN 3 TRIA 1 TERM
 0 0  1 0  0 1
 1 2 3
COMP COUL VERT LECT tous TERM
MATE LINE RO 8000.0 YOUN 1.E12 NU 0.0
          LECT tous TERM
OPTI PINS DUMP
          STAT
          VIDE
         !EQVL
         !EQVD
          ASN
LINK COUP
     PINB BODY MLEV 2 LECT tous TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA CONT ECRO FREQ 1
     FICH ALIC FREQ 1
OPTI NOTE
 CSTA 0.5E0
 LOG 1
CALCUL TINI 0. TEND 100.E-3
*=================================================================
PLAY
CAME   1 EYE   5.00000E-01  5.00000E-01  6.12372E+00
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB PARE NASN PASN
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB CDES DASN
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
ENDPLAY
*=================================================================
FIN
```

vide03.epx

```
VIDE03
ECHO
 CONV win
CPLA LAGR
GEOM LIBR POIN 2 ED01 1 TERM
 0 0  1 1
 1 2
COMP COUL VERT LECT tous TERM
MATE FANT 8000.0 LECT tous TERM
OPTI PINS DUMP
          STAT
          VIDE
         !EQVL
         !EQVD
          ASN
LINK COUP
     PINB BODY MLEV 2 LECT tous TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA CONT ECRO FREQ 1
     FICH ALIC FREQ 1
OPTI NOTE
 CSTA 0.5E0
 LOG 1
CALCUL TINI 0. TEND 100.E-3
*=================================================================
FIN
```

vide04.epx

```
VIDE04
ECHO
 CONV win
CPLA LAGR
GEOM LIBR POIN 1 PMAT 1 TERM
 1 1
 1
COMP COUL VERT LECT tous TERM
     EPAI 0.1 LECT 1 TERM
MATE FANT 8000.0 LECT tous TERM
OPTI PINS DUMP
          STAT
          VIDE
         !EQVL
         !EQVD
          ASN
LINK COUP
     PINB BODY DIAM 1.0 LECT tous TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA CONT ECRO FREQ 1
     FICH ALIC FREQ 1
OPTI NOTE
 CSTA 0.5E0
 LOG 1
CALCUL TINI 0. TEND 100.E-3
*=================================================================
FIN
```

vide05.epx

```
VIDE05
ECHO
!CONV win
TRID LAGR
GEOM LIBR POIN 8 CUBE 1 TERM
 0 0 0  1 0 0  1 1 0  0 1 0
 0 0 1  1 0 1  1 1 1  0 1 1
 1 2 3 4 5 6 7 8
COMP COUL VERT LECT tous TERM
MATE LINE RO 8000.0 YOUN 1.E12 NU 0.0
          LECT tous TERM
OPTI PINS DUMP
          STAT
          VIDE
         !EQVL
```

```
         !EQVD
          ASN
LINK COUP
     PINB BODY MLEV 2 LECT tous TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA CONT ECRO FREQ 1
     FICH ALIC FREQ 1
OPTI NOTE
 CSTA 0.5E0
 LOG 1
CALCUL TINI 0. TEND 100.E-3
*==================================================================
PLAY
CAME   1 EYE  -1.92464E+00 -3.79261E+00  6.15196E+00
!        Q     9.19967E-01  3.32438E-01 -1.12640E-01 -1.74520E-01
         VIEW  3.23285E-01  5.72348E-01 -7.53595E-01
         RIGH  9.13710E-01 -3.95997E-01  9.12164E-02
         UP    2.46214E-01  7.18056E-01  6.50980E-01
         FOV   2.48819E+01
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB PARE NASN PASN
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB CDES DASN
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
ENDPLAY
*==================================================================
FIN
```

vide06.epx

```
VIDE06
ECHO
 CONV win
TRID LAGR
GEOM LIBR POIN 4 Q4GS 1 TERM
 0 0 0  1 0 0  1 1 0  0 1 0
 1 2 3 4
COMP EPAI 0.1 LECT tous TERM
     COUL VERT LECT tous TERM
MATE LINE RO 8000.0 YOUN 1.E12 NU 0.0
          LECT tous TERM
OPTI PINS DUMP
          STAT
          VIDE
         !EQVL
         !EQVD
          ASN
LINK COUP
     PINB BODY MLEV 0 LECT tous TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA CONT ECRO FREQ 1
     FICH ALIC FREQ 1
OPTI NOTE
 CSTA 0.5E0
 LOG 1
CALCUL TINI 0. TEND 100.E-3
*==================================================================
FIN
```

vide07.epx

```
VIDE07
ECHO
!CONV win
CPLA LAGR
GEOM LIBR POIN 6 CAR1 2 TERM
 0 0  1 0  2 0
 0 1  1 1  2 1
 1 2 5 4
 2 3 6 5
COMP COUL VERT LECT tous TERM
MATE LINE RO 8000.0 YOUN 1.E12 NU 0.0
          LECT tous TERM
OPTI PINS DUMP
          STAT
          VIDE
         !EQVL
         !EQVD
          ASN
LINK COUP
     PINB BODY MLEV 2 LECT tous TERM
ECRI COOR DEPL VITE ACCE FINT FEXT FLIA CONT ECRO FREQ 1
     FICH ALIC FREQ 1
OPTI NOTE
 CSTA 0.5E0
 LOG 1
CALCUL TINI 0. TEND 100.E-3
*==================================================================
PLAY
CAME   1 EYE   1.00000E+00  5.00000E-01  7.83758E+00
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB PARE NASN PASN
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
SCEN GEOM NAVI FREE
          FACE HFRO
          PINB CDES DASN
     COLO PAPE
SLER CAM1 1 NFRA 1
TRAC OFFS FICH BMP REND
ENDPLAY
*==================================================================
FIN
```

List of input files

European Commission
EUR 26714 EN – Joint Research Centre – Institute for the Protection and Security of the Citizen

Title: Implementation of Assembled Surface Normals and of a Penalty Contact Formulation in the Pinball Model of EUROPLEXUS

Authors: Folco Casadei, Martin Larcher, Georgios Valsamos, Vincent Faucher

Luxembourg: Publications Office of the European Union

2014 – 212 pp. – 21.0 x 29.7 cm

EUR – Scientific and Technical Research series – ISSN 1831-9424

ISBN 978-92-79-38984-9
doi:10.2788/88194

Abstract

The most popular contact algorithms available in finite element computer codes are probably the so-called slide line (in 2D) and slide surface (in 3D) algorithms proposed by Hallquist and Benson. They are based on the notion of penetration of slave nodes into master segments (in 2D) or into master surfaces (in 3D). These algorithms suffer from a number of geometrically pathological cases in which physical penetration is not detected. The pinball method proposed by Belytschko and co-workers from the late 80's for application in impact problems with penetration is much more robust as concerns penetration detection. The pinball contact-impact method has been implemented in EUROPLEXUS, initially based upon a strong, Lagrange-multiplier based solution strategy of the contact constraints. Recently, the so-called Assembled Surface Normal (ASN) algorithm of Belytschko and an alternative penalty-based solution of the contact constraints have also been introduced as an option in the code. They are described in the present report.

LB-NA-26714-EN-N

JRC Mission

As the Commission's in-house science service, the Joint Research Centre's mission is to provide EU policies with independent, evidence-based scientific and technical support throughout the whole policy cycle.

Working in close cooperation with policy Directorates-General, the JRC addresses key societal challenges while stimulating innovation through developing new methods, tools and standards, and sharing its know-how with the Member States, the scientific community and international partners.

Serving society
Stimulating innovation
Supporting legislation

Publications Office

ISBN 978-92-79-38984-9
doi:10.2788/88194